Lattice Boltzmann simulation methods for boundaries and interfaces in multi component flow.

HOLLIS, Adam P.

REFERENCE

Lattice Boltzmann Simulation Methods for Boundaries and Interfaces in Multi Component Flow.

Adam Peter Hollis.

A thesis submitted in partial fulfilment of the requirements of Sheffield Hallam University for the degree of Doctor of Philosophy

Materials Research Institute, Sheffield Hallam University, Howard Street, S1 1WB, UK

February 2009

Abstract

In this work I shall give details of the development of a two dimensional Lattice Boltzmann algorithm targeting the simulation of immiscible fluids at low Reynolds number and low Capillary number. The Lattice Boltzmann method will be explained along with its multi–component extensions. Key method developments shall first be developed in terms of single component advancements, made in the lattice closure algorithm and in the application of external forces to boundary lattice sites. Having secured single component advancements, I shall present parallel developments made in immiscible flow simulation, considering two immiscible fluids (however extension to larger numbers of immiscible species is mentioned where appropriate). Drop dynamics within shear flow shall be examined with Numerical colour segregation along with attempts for the application of a kinematic condition. Analytic segregation shall then be used and the dynamics of the phase field shall be analysed showing improved drop dynamics. A simple and adaptable method for application of a kinematic condition shall next be shown to be effective when used in conjunction with the analytic diffusion method in improving the quality of the models hydrodynamics. Culmination of all the previously identified improvements to the simulation method shall then be utilised in the simulation of wetting drops in both static and dynamic situations. The final method is qualitatively shown to predict static wetting, rolling contact points, bifurcating contacts and the spreading of films.

Acknowledgments

I would like to express my gratitude to my supervisory team Dr. Ian Halliday who inspired me throughout this work and my undergraduate studies and Prof. Chris Care for their assistance and guidance through the course of this work.

Fluent Europe Ltd to whom I am very grateful has provided funding for my research.

I would like to thank my external examiner Dr Rammille Ettelaie and my internal examiner Prof Marcos Rodrigues who very kindly agreed to be my examiner at short notice.

I would finally like to thank my family but especially my parents Margaret and Peter Hollis for their continued support, both financially and emotionally, and instilling me with the confidence and drive to complete this work.

Publications

Below is a list of publications to date:

- I. Halliday, R. Law, C.M. Care and A. Hollis. Improved simulation of drop dynamics in a shear flow at low Reynolds and capillary number. Phys. Rev. E, **73**, 056708, (2006).

- A. Hollis, I. Halliday and C.M. Care. Enhanced, mass-conserving closure scheme for lattice Boltzmann equation hydrodynamics. J. Phys. A: Math. Gen. **39**, 10589 - 10601, (2006).

- I. Halliday, A. Hollis and C.M. Care. Lattice Boltzmann algorithm for continuum multi–component flow. Phys. Rev. E, **76**, 026708, (2007).

- I. Halliday, A. Hollis and R. Law . Kinematic condition for multi–component lattice Boltzmann simulation. Phys. Rev. E, **76**, 026709, (2007).

- A. Hollis, I. Halliday and C.M. Care. An accurate and versatile lattice closure scheme for lattice Boltzmann equation fluids under external forces. J.Comp. Phys. **277**, 17, (2008)

- Lawford P.V, Ventikos Y, Khir A.M, Atherton M., Evans D, Hose D.R, Care C.M, Watton, P.N, Halliday I, Walker D.C, Hollis A.P, Collins M. Modelling the Interaction of Haemodynamics and the Artery Wall: Current Status and Future Prospects. Biomedicine and Pharmacotherapy. **62**, 8, (2008).

Contents

1 Thesis Overview

Work carried out as part of this report has been supported by Fluent Europe Ltd. Although some of the nomenclature and language used in this section are yet to be explained; it is deemed appropriate at the outset to provide an overview of the aims and objectives of the work and to outline the intended route to fulfilling these. This section shall detail the initial issues associated with the inherited multi–component lattice Boltzmann method and the steps taken to overcome these issues. The layout of the thesis shall also be described.

1.1 Aims of thesis

At the beginning of this work the multi–component lattice Boltzmann (MCLB) method showed great potential in a number of important areas of continuum flow calculations. However there remained a variety of significant problems within the core method, see Chapter 7 for an in-depth analysis of the associated problems with the core method.

The major issues that are to be addressed within this work may be summarized as follows:

1. Micro-currents on multi–component interface obscuring hydrodynamic signals.
2. Lattice pinning in low Reynolds number flows or "sticking to the simulation lattice" of the multi–component interface.
3. Indeterminate interfacial hydrodynamics and a verifiable absence of a kinematic condition.
4. Verifiable contact line behaviour at an accurately represented boundary with the ability to recover Navier conditions.

It is worth noting that the final item decomposes into a number of sub-issues that are to be addressed separately.

The intended aim of this project was to improve and develop lattice Boltzmann simulation of multi–component immiscible fluids, with particular attention being paid to the above issues, thus facilitating simulations in micro-fluidic regimes and venule scale blood flow applications. It is stressed at the outset that individual application of the method to a specific situation is not the purpose of this work. The ultimate aim is to produce an effective and accurate LB method for the simulation of interface-dominated flow including wetting in both static and dynamic conditions.

1.2 Layout

Considering the issues arising in the inherited model that were identified in the previous section; it is clear that the work devolves into two sections.

1. Consideration of boundary conditions.

2. Fundamental issues arising from MCLB interfaces.

In terms of the structure of the thesis I intend to address the above issues as follows:

I initially intend to address the issue of boundary conditions exhaustively. For tractability and for my own personal introduction into the subject it was decided that boundary conditions are, initially, to be examined using single component lattice Boltzmann (SCLB).

Having resolved any issues arising while working on an accurate and effective boundary condition; the remaining problems that are associated with the MCLB interface shall be resolved.

Finally the two work-packets listed above shall be combined in order to create an effective algorithm for the treatment of multi–component fluids at solid boundaries. Having done this the issues raised in items 1–4 in 1.1 above will have been resolved.

Considering the above discussion the structure of this thesis is as follows:

Following an introduction in Chapter 2; Chapters 3 and 4 *highlight* all relevant background material used in subsequent chapters. Relevant fluid mechanics is briefly

discussed in Chapter 3 with relevant lattice methods reviewed in chapter 4. It is noted here that this is not intended to be a complete treatment of fluid mechanics but a brief overview intended to put the work into context.

Chapters 5 and 6 contain a rigorous treatment of the issues arising with boundary conditions. This treatment of boundary closure methods is carried out using SCLB initially. It will be generalized later in Chapter 10.

Chapter 7 details my initial attempts using an inherited MCLB. The issues highlighted in points 1–3 in 1.1 are addressed with only partial success. This chapter serves to highlight the inadequacies of the core model that I inherited.

Chapters 8 and 9 contain a fundamental revision to the core MCLB algorithm which proved successful in overcoming the issues of 1–3 in 1.1.

Chapter 10 attempts to combine advances made in SCLB from chapters 5 and 6 with the MCLB advances made in chapters 8 and 9. The aim of this chapter is to show how a flexible method has been developed for boundary closure, that is capable of dealing effectively with interfaces and Navier slip conditions.

Chapter 11 proposes future work and extensions to the new methods and suggests possible methods of improvement.

2 Introduction

Fluid dynamics is a branch of physics and mechanics dealing with fluids, these can be liquid i.e. hydrodynamics or gaseous i.e. aerodynamics. The effects of fluid motion impacts upon us within most areas of our daily lives from aerodynamics within automotive industries to pharmaceutical manufacture ensuring even distribution of components drug mixing within crucibles. A conventional approach within engineering environments is to recreate flow phenomena through the use of certain dimensionless numbers, that are discussed later. Using these dimensionless parameters it is possible to physically recreate various flow environments through 'dynamic similitude'. In other words, real systems may be classified by dimensionless numbers so as to recreate identical fluid effects using different physical models described by the same numbers. Such concepts are used in practical experiments such as wind tunnels etc. Physical scaling of systems is an effective way of reproducing fluid phenomena within flow domains however there are also various disadvantages to dynamic similitude. Conducting physical dynamic similitude experiments is a time consuming and costly process, making minor changes to the flow domain can be difficult, quantitative data may only be collected from discrete points within the simulation domain giving local information only. Additional complications arise when conducting dynamic similitude experiments when various flow regimes are to be analysed for example laminar or turbulent flow, difficulties can arise in producing identical experimental conditions for both flow regimes. Computational fluid dynamics packages therefore offer a cost effective, transportable and adaptable way of simulating systems to aid understanding of physical experiment or to simulate systems that may be too complex for dynamic similitude experiments. In conjunction with theoretical fluid mechanics, the three methods allow us to better understand the processes taking place within flow systems.

In respect of the computation methods at the heart of this thesis, the first at-

tempt to model fluid flow successfully and recover the rotational symmetry of the Navier–Stokes equations was made in 1986 by Uriel Frisch, Brosl Hasslacher and Yves Pomeau (FHP) [1] when they produced a cellular automaton (CA) computer simulation that worked by obeying conservation laws to recover hydrodynamics on a microscopic level. This early CA worked using a two step mechanism, a free-stream and a collision; the free-stream operation involves the movement of Boolean particles from one site on a hexagonal lattice to an adjacent lattice site, thus making the method a Lattice Gas Cellular Automation (LGCA). The collision step occurs when Boolean particles reach the same site on the lattice at the same time and interact; the total momentum is conserved but redistributed between the particles at that site. The redistribution is governed by a set of local collision protocols. This system is clearly time-space discretised.

Due to the large amounts of integers being processed for a Boolean LGCA simulation there is inevitably large amounts of statistical noise. In an effort to try and reduce the levels of statistical noise and recover behaviour more conclusively related to Navier–Stokes behaviour, the Boolean components were replaced with, effectively, an ensemble-average population based on a distributed Maxwell–Boltzmann distribution. It was soon realized that the technique of replacing the Boolean particles and integer arithmetic with floating point arithmetic on continuous populations was a valid method of reproducing fluid hydrodynamics in its own right [2]. Hence the first form of the Lattice Boltzmann Equation (LBE) was produced.

Fluid mechanics relies on the laws of conservation, in this report the Lattice Bhatnagar Gross and Krook (LBGK) algorithm will be used for the implementation of the Boltzmann equation on a lattice. The algorithm is implemented on a two dimensional lattice with nine possible velocities (D2Q9) that will be covered in more detail later. [3] [4]

Fluids considered in this work are Newtonian fluids in the continuum regime. That is, that stress and strain within the fluid are linearly related through their

viscosity. The fluids are viewed in the continuum regime or in other words are considered to be continuously divisible. The interaction of the individual particles comprising the fluid are not directly considered. Thus physical parameters such as the velocity of the fluid and the fluid density may be specified at any point within the flow domain.

Throughout this work, flow shall be regarded as incompressible and isothermal. All fluids are compressible to some degree however the assumption of incompressibility allows us to assume a constant density throughout the continuum fluid and thus a uniform viscosity. As the fluids are considered to be isothermal we assume them to have a uniform temperature throughout. Consideration of isothermal and incompressible fluids allows us to greatly simplify the equations governing flow. There has been substantial work carried out considering the energy equation that governs flow [5], as we are considering isothermal and incompressible fluids, this shall not be considered.

The law of mass conservation will be used to derive the Continuity equation and the conservation of momentum will be shown to give rise to a set of three equations known as the Navier–Stokes equations. The Continuity and Navier–stokes equations can be used to describe the behaviour of any fluid within a flow domain, the unique characteristics of the flow are a consequence of the interaction of the fluid with domain boundaries and interface interactions between immiscible components.

Accurate representation of the domain boundaries including the interface between two immiscible fluids is therefore essential if accurate hydrodynamics are to be recovered.

2.1 Recent Related Advances Using Lattice Boltzmann

Throughout this work various recent developments in LBE modelling are directly utilised and are referenced where appropriate such as M.Latva-Kokko and D.H.

Rothman's segregation method [6] which was published in 2005. In order to illustrate and contextualize recent developments and to show wider applications of the method I would like to give a brief overview of recent work carried out by O. Berk Usta and Alexander Alexeev from the Chemical Engineering Department of the University of Pittsburgh [7]. The following application illustrates the broader context of the detailed work undertaken in this thesis.

Usta's work builds on earlier simulation techniques of Alexander Alexeev *et al* [8]. Alexeev uses two different meso-scale models and integrates them in order to simulate the behaviour of micro-capsules in a continuum fluid. The aim of Alexeev's work is to combine an LB method fairly similar to ours and a lattice spring method [9] (which will not be discussed in this work), to simulate the behaviour of a continuum fluid alongside a micro-capsule's elastic shell. The work aims to improve the speed at which chemical data can be retrieved from micro-reactors. The aim is to manipulate minute quantities of reagents that are contained within the micro-capsules. Alexeev also notes that the fluid-filled elastic shelled micro-capsules can also be used as simple models for biological cells such as leukocytes.

Using similar boundary conditions to those developed by Alexeev*et al* [8] described above; Usta has recently published work describing how, by using the combined LB and lattice spring method of Alexeev, they have been able to simulate the self propelled motion of micro-capsules.

Usta describes how in some biological systems there is communication between cells which are designated signalling and target cells, this allows the cells to cooperate and thus carry out a large range of functions. The aim of Usta's work is to mimic the communication that occurs between biological cells. The main outcome is that when considering a pair of cells, one signalling and one target, that are situated in close proximity on a substrate, the action of the signalling cell results in the self-propelled motion of both cells.

Initially consider a single signalling cell on a substrate that is surrounded by a

continuum fluid. The signalling cell is filled with dispersed nano-particles within its fluid filled core. These nano-particles are able to diffuse through the shell of the micro-capsule. The nano-particles can "chemisorb" onto the surface of the substrate and modify its wetting properties, in particular its adhesive interaction. The strength of adhesion of the substrate diminishes as the concentration of nano-particles chemisorbed onto it increases. The signalling cell therefore emits nano-particles that diffuse through the surrounding fluid according to Brownian motion. The deposition of nano-particles on the substrate creates an adhesion gradient emanating from the point directly underneath the centre of mass of the signalling cell. At this stage Usta notes that there is no self-propelled motion as the system is symmetrical.

Placing a target cell in close proximity to the signalling cell however breaks the system's symmetry. Upon diffusion of the nano-particles, the wetting section of the target cell is exposed to a adhesion gradient. Should the parameter set be appropriate in terms of diffusion rates and relative separation and reactivity of the nano-particles relative to the adhesivness of the surface; the target cell will, under the correct circumstances, move from the less adhesive area to the more adhesive area, i.e. along the adhesion gradient.

Subsequent to the initial movement of the target cell, Usta found that the hydrodynamics of the continuum fluid caused by the motion of the target cell, initiated movement of the signalling cell itself thus following the target cell along the adhesion gradient. Following this initial movement, both the target cell and the signalling cell lie in asymmetric positions relative to the adhesion gradient and thus continue to move provided that the signalling cell continues to diffuse nano-particles.

Therefore, by utilising the lattice Boltzmann method, Usta *et al* have designed a biometric system where communication is achieved between a pair of synthetic particles through the diffusion of nano-particles. They have also shown how the hydrodynamics induced by the target cell can influence the motion of the signalling

cell thus demonstrating that hydrodynamic forces have an important role in the sensing and signalling behaviour of biological systems.

I will not attempt simulation with the degree of complexity of Usta *et al*'s [7] work; my objective is to examine the fundamental accuracy and verifiability of the underlying LB model. However, Usta's work does illustrate the complexity of the fluid systems that LB can address. The following is the sort of thing that the models I have developed might, eventually, be applied to.

There is a long history of modelling attaching cells as wetting immiscible drops. In the context of this thesis, a similar result to that obtained by Usta *et al* may be achieved by using the culmination of advances described in this thesis, shown in chapter 10. If simulations of this kind were to be carried out, the wetting properties of the substrate, specifically in the above example, the adhesion properties of the substrate would need to be investigated. In addition, the secondary migration of the signalling cell that is induced by the hydrodynamics of the continuum fluid must not be compromised by the micro-currents of the interface. The low micro-current interface algorithm demonstrated in Chapter 10 is necessary.

The essential difference between the work in this thesis and that of Usta *et al* is our focus on hydrodynamic boundary conditions and their accurate correct representation without recourse to any form of (computationally expensive) and unverified membrane model.

3 Introduction to Fluid Mechanics

The intention here is not to cover fluid dynamics in a comprehensive manner however it is appropriate to consider some key concepts in context; more complete accounts are to be found in Batchelor [10], Tritton [11], Landau and Lifshitz [5] and Happel and Brenner [12].

3.1 Dimensionless Quantities in Hydrodynamics

In fluid mechanics it is conventional to describe various flow regimes using a variety of different dimensionless quantities. It is appropriate at this stage to mention some of these dimensionless quantities and their meanings. There are a wide range of dimensionless numbers that are applicable to a large variety of flow situations such as; the Nusselt Number (that measures the enhancement of heat transfer within a fluid when convection is considered as well as conduction) and the Schmidt Number (that is used to describe the ratio of diffusion of momentum to that of mass). Dimensionless quantities that are to be used regularly in this work are explained below in greater detail.

Throughout this work the following standard notation shall be employed. The shear and kinematic viscosity are denoted as η and ν respectively. The drops we consider have radius R and surface tension α. Their flow environment is usually characterised by a velocity U_0 and/or a shear rate $\dot{\gamma}$.

3.1.1 Capillary Number

In fluid mechanics the capillary number (Ca) is used when considering the interface between two immiscible fluids or liquid-gas interfaces. The Ca measures the ratio between the effects of viscous forces relative to the surface tension acting across the interface. The Ca may be regarded as the ratio of momentum diffusion forces to

surface tension forces. The Ca may be expressed quantitatively as follows:

$$Ca = \frac{\nu U_o}{\alpha}. \tag{1}$$

Where all symbols have meanings as described previously.

3.1.2 Mach Number

The Mach number (Ma) of a flow may be regarded as a measure of the speed of the flow. It is defined as the ratio of the speed of an object relative to the fluid and the speed of sound with in the fluid medium. The Ma may be expressed as follows:

$$Ma = \frac{U_o}{c_s}. \tag{2}$$

Where c_s is the speed of sound within the medium and all other symbols have their usual meaning as described previously.

It is worth noting that the Mach number is temperature dependant as the speed of sound within a given medium varies with temperature. When considered in the context of this work however the Ma may be considered constant as we are solely concerned with isothermal flows.

The Ma is important when considering fluid mechanics simulations as solids being passed by a fluid at the same Ma will experience similar forces and effects even though the velocity of the passing fluid may not be identical in both cases.

3.1.3 Reynolds Number

The Reynolds number (Re) of a flow practically quantifies the isothermal flow regime and is used to distinguish between various types of flow for example laminar and turbulent flows. The Re is considered the fundamental and most important dimensionless number used in fluid mechanics. The Re is defined as the ratio of the inertial

forces in flow to the viscous forces. It may be expressed as the bulk Reynolds number using the dimension of the flow domain or may be expressed as the 'drop Reynolds number' (Re_d) which uses the local dimension of a drop contained within flow as follows;

$$Re = \frac{U_o W}{\nu}, \tag{3}$$

$$Re_d = \frac{\dot{\gamma} R^2}{\nu}, \tag{4}$$

where W is the physical dimension of the flow domain, $\dot{\gamma}$ is the shear rate, R is the radius of the drop and ν is the dynamic viscosity of the fluid. As with the Mach number, the Reynolds number may be used in conjunction with other dimensionless numbers to calibrate simulations of actual systems where dynamic similitude relates different experiments.

3.2 Governing Equations of Hydrodynamics

3.2.1 Continuity Equation

The Continuity Equation arises as a consequence of the principle of conservation of mass within a flow domain. To illustrate this we can consider a control surface element of surface area, dA, and its normal unit vector $\hat{n}$. If we consider the amount of fluid passing through the surface area in a unit time with a velocity $\underline{u}$ we have:

$$\underline{u}.\hat{n}dA = \underline{u}.\underline{dA}; \quad \underline{dA} \equiv \hat{n}dA. \tag{5}$$

Multiplying the above expression through with the density of the fluid gives us an expression for the mass of fluid passing through the surface element in a unit time. Now using surface integration techniques and applying the law of mass conservation, the total change in mass of the volume, v_0, with surface area, A, must equal the

amount of mass passing through the surface, s, in a unit time:

$$\frac{\partial}{\partial t}\int\int\int_{v_0}\rho.du = -\int\int_{s}\rho\underline{u}.dA. \tag{6}$$

Using the divergence theorem on the reference volume provided that mass is neither created nor destroyed, we have;

$$\int\int\int_{v_0}\frac{\partial\rho}{\partial t}.du = \int\int\int_{v_0} -\underline{\nabla}(\rho\underline{u})du. \tag{7}$$

Equating integrals we obtain from the principle of mass conservation and the divergence theorem [13];

$$\frac{\partial\rho}{\partial t} + \underline{\nabla}(\rho.\underline{u}) = 0. \tag{8}$$

We have therefore shown that the continuity equation comes as a direct result of the conservation of mass within a system. The continuity equations is the most fundamental equation within fluid mechanics.

3.2.2 Euler Equation

The Navier-Stokes equations arise from the conservation of momentum and the application of Newton's second law of motion. The equations are parabolic second–order partial differential equations that are fundamental in fluid mechanics and, along with the continuity equation and the flow boundary conditions, close the description of hydrodynamic flow processes.

Prior to derivation of the Navier Stokes equations it is appropriate to derive the Euler equation. Which applies to the ideal and reversible flow processes.

Force exerted by a liquid on a material enclosed in a surface "s" is :

$$\underline{F} = -\int\int_{S} P\underline{dA}. \tag{9}$$

Now, by applying the divergence theorem to the vector $\underline{V} = V\hat{a}$ with $\hat{a}$ a constant unit vector, it is straightforward to show that:

$$\int\int_S \underline{V}.\underline{dA} = \hat{a}.\int\int_s V\underline{dA} = \hat{a}.\int\int\int_V \nabla V d^3\underline{r}, \tag{10}$$

that is;

$$\int\int_S V\underline{dA} = \int\int\int_V \nabla V d^3\underline{r}, \tag{11}$$

which when applied to equation (9) gives;

$$\underline{F} = -\int\int_S P.\underline{dA} = -\int\int\int_{v_0} (\nabla P) d^3\underline{r}. \tag{12}$$

Making the definition for the force per unit volume on the immersed reference body to be:

$$\underline{F} = \int\int\int_{v_0} \underline{F'} d^3\underline{r}, \tag{13}$$

which may be inserted into equation (12) in order to give the equivalence;

$$\underline{F'} = -\nabla P. \tag{14}$$

Consider Newton's second law of motion and express it as a material derivative the momentum of the unit volume. Expanding the material derivative we may express the force per unit volume in terms of time and space derivatives:

$$\underline{F'} \equiv \rho\frac{D}{Dt}\underline{u} = \rho(\frac{\partial \underline{u}}{\partial t} + u_x\frac{\partial}{\partial x}\underline{u} + u_y\frac{\partial}{\partial y}\underline{u} + u_z\frac{\partial}{\partial z}\underline{u}). \tag{15}$$

Compressing the spatial derivatives and making use of the equivalence identified in (14) we can write the Euler equation:

$$-\nabla P = \rho(\frac{\partial \underline{u}}{\partial t} + (\underline{u}.\nabla).\underline{u}). \tag{16}$$

We have derived the Euler equation for an incompressible fluid but it remains true for a compressible fluid also.

3.2.3 Navier-Stokes Equation

We may now proceed to derive the Navier-Stokes equations. Initially we can denote the Euler equation (16) in cartesian tensor form and rearrange with the view of obtaining a form into which we can insert the momentum flux tensor, Π_{ik} [5], and defined below. Taking the Euler equation in tensor notation and rearranging we obtain;

$$\frac{\partial(u_i)}{\partial t} = -(u_k \frac{\partial u_i}{\partial x_k}) - \frac{1}{\rho}\frac{\partial P}{\partial x_i}. \tag{17}$$

If we now consider separately the time differential $\frac{\partial(\rho u_i)}{\partial t}$ and expand the derivative using the chain rule we obtain a two term differential equation, if we proceed to replace the density differential on time with the identity proposed by the Continuity equation we can obtain an expression;

$$\frac{\partial u_i}{\partial t} = \frac{1}{\rho}\frac{\partial(\rho u_i)}{\partial t} + \frac{u_i}{\rho}\frac{\partial(\rho u_k)}{\partial x_k}. \tag{18}$$

Substituting (18) into (17) and rearranging we obtain,

$$\frac{1}{\rho}\frac{\partial(\rho u_i)}{\partial t} + \frac{u_i}{\rho}\frac{\partial(\rho u_k)}{\partial x_k} = -u_k\frac{\partial u_i}{\partial x_k} - \frac{1}{\rho}\frac{\partial P}{\partial x_i}. \tag{19}$$

Rearranging (19) allows us to condense derivatives in x_k using the reverse of the product rule for differentiation giving;

$$\frac{\partial \rho u_i}{\partial t} = -\frac{\partial P}{\partial x_i} - \frac{\partial \rho u_k u_i}{\partial x_k}. \tag{20}$$

The free variable tensor subscripts may be changed using the property of the Kronecker delta function, the resulting equation may then be factorized producing an

equation that will allow us to identify a momentum flux tensor function:

$$\frac{\partial \rho u_i}{\partial t} = -\frac{\partial}{\partial x_k}(P\delta_{ik} + \rho u_k u_i), \tag{21}$$

defining;

$$\Pi_{ik} = P\delta_{ik} + \rho u_k u_i, \tag{22}$$

we have;

$$\frac{\partial \rho u_i}{\partial t} = -\frac{\partial}{\partial x_k}\Pi_{ik}. \tag{23}$$

The momentum flux tensor Π_{ik} is defined as the i^{th} component of the momentum transferred into a reference volume u_0 through a surface, S with its normal in the x_k direction in a single unit time. It can be noted that the current form of the Euler equation is applicable to ideal fluids only as it represents a momentum transfer mechanism that is totally reversible.

In order to obtain the appropriate hydrodynamics for a non-ideal, viscous fluid, internal friction must be incorporated into the Euler equation (where friction is regarded as dissipation of kinematic energy) which has been expressed in equation (23) in terms of the momentum flux tensor. In order to take account of friction effects the momentum flux tensor is modified, incorporating an additional term, the viscous stress tensor σ'_{ik}. This term introduces momentum dissipation into the momentum transfer mechanism of the fluid. The modified momentum flux tensor may be expressed as;

$$\Pi_{ik} = \Pi_{ik} - \sigma'_{ik}. \tag{24}$$

Inserting the modified form of the momentum flux tensor into (22) and substituting in for Π_{ik} in (23) it is possible to factorize the resulting equation into the following form;

$$\frac{\partial \rho u_i}{\partial t} = \frac{\partial}{\partial x_k}(-P\delta_{ik} + \sigma'_{ik}) - \frac{\partial}{\partial x_k}\rho u_i u_k. \tag{25}$$

The term, $-P\delta_{ik} + \sigma'_{ik}$, in (25) is the full stress tensor, σ_{ik}, and $-P\delta_{ik}$ is known as the hydrodynamic stress component and, as we have discussed, σ'_{ik}, is the viscous stress tensor.

From physical arguments the viscous stress tensor must become zero for a fluid that is under uniform translational or rotational motion. From such physical arguments it is possible to derive an expression for the viscous stress tensor. Such a derivation is carried out in [5] with the result being;

$$\sigma'_{ik} = \eta(\frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i} - \frac{2}{3}\delta_{ik}\frac{\partial u_l}{\partial x_l}) + \xi(\frac{\partial u_l}{\partial x_l})\delta_{ik}. \tag{26}$$

Where η and ξ are coefficients of viscosity and are both positive and functions of pressure and temperature. As we are only considering isothermal and incompressible flows, these coefficients are constants. η is known as the shear viscosity and ξ is the bulk viscosity. Substituting into equation (25) and making use of the Kronecker delta function we can express the expanded form of the Euler equation with additional retardation as;

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_k u_i}{\partial x_k} = -\frac{\partial P}{\partial x_i} + \eta\nabla^2 u_i + (\xi - \frac{2}{3})\frac{\partial}{\partial x_i}\frac{\partial u_l}{\partial x_l}. \tag{27}$$

Using standard algebraic techniques the left hand side of equation (27) may be expanded using the product rule, the resulting equation may then be factorized and expressed in a form that allow the continuity equation to be used to cancel terms and simplify the expression:

$$\rho(\frac{\partial u_i}{\partial t} + u_k\frac{\partial u_i}{\partial x_k}) = -\frac{\partial P}{\partial x_i} + \eta\nabla^2 u_i + (\xi + \frac{1}{3}\eta)\frac{\partial}{\partial x_i}(\frac{\partial u_l}{\partial x_l}). \tag{28}$$

The fact that the kinematic viscosity is given by $\frac{\eta}{\rho} = \nu$, also from the Continuity equation, for an incompressible fluid $\frac{\partial u_k}{\partial x_k} = 0$. It is therefore possible to produce and

expression for the incompressible Navier-Stokes:

$$\frac{\partial u_i}{\partial t} + u_k \frac{\partial u_i}{\partial x_k} = -\frac{1}{\rho}\frac{\partial P}{\partial x_i} + \nu \nabla^2 u_i. \tag{29}$$

The incompressible Navier-Stokes equation and the Continuity Equation form the most important set of equations in hydrodynamics.

3.3 Initial Conditions

Chapter 5 deals with boundary conditions in the context of lattice Boltzmann simulation in detail. It is appropriate, however, to make a few remarks here.

We consider only incompressible fluids with the equation of state $\rho = constant$; compressible fluids need a little more care.

The Navier–Stokes equation is a parabolic partial differential equation. To close a particular solution it is therefore necessary to have (i) its solution (the velocity field) specified over the complete, closed perimeter of the flow domain at all times and (ii) an initial condition specified over the whole flow domain. Note that it is not necessary to know the pressure distribution over the boundary. (In principle, pressure may be eliminated from the description by taking the curl of the Navier–Stokes equation, which may then be solved. Pressure is later derived from the known velocity by solving a Poisson–type equation derived by taking the divergence of the Navier–Stokes equation and using the continuity equation).

In cases where the steady–state of flow is alone of interest, the elliptic time–independent Navier–Stokes equation must be solved with reference to the velocity field specified over the complete, closed perimeter of the flow domain.

3.4 Interface Conditions

It may be shown that from mechanical stability considerations, namely Newton's third law of motion, that at an interface between two immiscible fluids (a) and (b),

characterized by a local normal $\hat{n}(r)$, the following stress conditions must hold for the viscous stress and the full stresses respectively:

$$\sigma_{ik}^{\prime(a)}\hat{t}_k - \sigma_{ik}^{\prime(b)}\hat{t}_k = 0, \tag{30}$$

$$\sigma_{ik}^{(a)}\hat{n}_k - \sigma_{ik}^{(b)}\hat{n}_k = \alpha(\frac{1}{R_1} + \frac{1}{R_2})n_i, \tag{31}$$

where $\hat{t}$ is the tangent to the interface ($\hat{t}.\hat{n} = 0$), R_1 and R_2 are the principal radii of curvature of the interface and α is the surface tension at the interface.

At this point it is important to note that (i) in the continuum approximation the interface enters the description of the problem of a boundary condition and (ii) that the interface is considered to have no structure.

3.5 Closing Remarks

It should be noted here that, as we are carrying out simulations using isothermal fluids, a large body of work that has been carried out concerning the scalar (partial differential) energy equation has been omitted. For information regarding the energy convection and diffusion (conduction) in fluids, the reader is directed to [5], for further information and derivation of the energy equation. It should also be noted that a closed mathematical description of a fluid in general requires the equation of state for the fluid to be specified. In the present context, however, lattice Boltzmann fluids have an ideal gas equation of state (see below) which simplifies the description of the lattice fluid (but introduces other problems because the speed of sound in such models is only $o(1)$). Notwithstanding the preceding comments, we shall work, henceforth, in the incompressible, isothermal regime of viscous flow, described by the continuity and Navier–Stokes equations.

It is also appropriate to remark on the large variety of analytical solutions to and simplifications of the flow problem. The range of analytical solutions and of the

methods and investigation strategies developed to solve the continuity and Navier Stokes equations are surveyed in a range of well–established texts, possibly the most significant of which is [10].

Physicists get away with approximating the greater part of many flows as "potential" (with velocity fields derived from the gradient of a scalar velocity potential), with magnetostatic analogies and a range of savage approximations of a less than rigorous nature. For more "physical insight" into such approaches to flow processes, and for an appreciation of the underlying flow physics, the reader is directed to [11], who describes fluid dynamics from the perspective of the underlying physics of the flow, in a way well suited for the less mathematically orientated reader and with reference to experimental facts.

4 Lattice Methods

Lattice methods have been a part of statistical mechanics for decades. Models such as the famous Ising model, was solved in two dimensions in 1944 by Onsager [14]. Lattice gas models for the simulation of fluids appeared in approximately 1976 as a result of work by Hardy [15]. With the progression of technology computational fluid dynamics came of age approximately a decade later. As previously discussed Lattice Boltzmann methods are a derivative of lattice gas cellular automaton. Here I shall present a brief summary of the methods used in this work in order of their algorithmic complexity; beginning with single component simulations and later progressing to multi–component simulations.

4.1 Single Component LB Methods

4.1.1 Background

Historically Lattice Boltzmann methods derived from Lattice Gas Cellular Automata. As mentioned in Chapter 2, one of the main problems arising in LGCA was noise in the simulations, which was overcome by using an ensemble average as a primary quantity as opposed to boolean quantities. Other problems with LGCA arise in the form of sensitivity to lattice structure; only certain choices of local lattice structure (i.e. the unit cell) retain sufficient symmetry in terms of their isotropy to allow successful recovery of Navier–Stokes behaviour [3]. Further problems arise when carrying out simulations in three dimensions, there are no unit cells that have sufficient symmetry to recover the appropriate hydrodynamics. In order to achieve three dimensional simulations with LGCA it is necessary to work in four dimensions to increase the isotropy of the system. This obviously increases the complexity of the simulation and requires large run times or the compilation of large collision rule tables, using large amounts of system memory.

Galilean invariance is not intrinsic in LGCA and the fluid velocity becomes pres-

sure dependant. When recovering the hydrodynamic equations that are Navier-Stokes like, it is found that using a discretised lattice velocity gives rise to a "Gallileance breaking" factor in the lattice fluids macroscopic equation. In order to remove this factor it was necessary to confine the simulation to low Mach number flows, thus making terms containing the Galileance breaking term negligible [16]. Further calculations must then be carried out in performing scaling operations of the velocity and time with the Galileance breaking term.

CA simulations are subject to further constraints in terms of the maximum Reynolds number (Re) that may be obtained. The maximum Re is governed by the number of collisions that may occur per time-step and is therefore dependent on the lattice used (due to the number of discrete velocities allowed). The possible number of allowed unit cell velocity vectors that are capable of conserving mass and momentum is therefore central in terms of achieving target flow regimes in simulation.

Having said all this however; what was previously considered as an inconvenience in CA has recently enjoyed a renaissance in CA; the calculation of actual collision sets in collision tables. Although inconvenient to encode, the boolean nature of the collision in LGCA simulations gives unconditional stability with no numerical divergence and is therefore being reinvestigated to apply the simulations to cases of turbulence.

LB began as a solution to the problem of statistical noise in the CA system by calculating time and space averaged quantities directly and using floating point populations as opposed to Boolean quantities. In recent times the lattice version of the Boltzmann equation (LBE) has become the most relevant derivative technique of the essential method, the model continues to attract interest above models like DPD [17–20] as the Boltzmann equation may be shown to recover the Navier-Stokes and the continuity equation through the Chapman-Enskog expansion to be considered later.

There are various forms of lattice Boltzmann equation simulation each with their individual merits for application to various situations. For example, in mesoscale, multi–component problems, where the kinematics of phase separation feature, the free-energy method [21, 22], based as it is upon Cahn–Hilliard theory, is the most appropriate choice of multi–component LB (MCLB). Alternatively, flows like that in reference [23], which are formally characterized as complex, incompressible, multi–component flows, address the continuum fluid approximation, to which we consider the LBGK (see below) method described here is most applicable.

In the latter applications, fluid-fluid interfaces are unstructured and appear as boundary conditions on $\partial\Omega_{12}$ where Ω_1 and Ω_2 are separate Navier Stokes domains. (Most MCLB variants have, at some time, been applied in this approximation but physical accuracy, efficiency and simplicity favour a MCLB pioneered by Gunstensen *et al* [24] and modified by Lishchuk [25] and a method of Latva-Kokko and Rothman [6] for the continuum regime).

The LB method used here is the LBGK method [4], arises as the BGK Boltzmann equation [26] is applied to a lattice and the Navier-Stokes and Continuity equations are recovered. The BGK Boltzmann equation is given in [26] as;

$$\frac{\partial}{\partial t}f + (\underline{U}.\nabla)f = -\frac{1}{\tau}(f - f_i^{(0)}), \tag{32}$$

where the link populations are given by the single particle population distribution function f_i and $f_i^{(0)}$ is the Maxwell-Boltzmann equilibrium distribution. $\underline{U}$ is the macroscopic velocity of the fluid and τ is the collision time. The term on the right hand side of equation (32) is the collision term, it is worth noting, that if the right hand side of the equation were to tend to zero, we have a situation where no collisions occur, as we would expect in a low density gas system. The collision term acts as to return the system to thermodynamic equilibrium, the further the distribution function (f_i) is away from the equilibrium distribution ($f_i^{(0)}$) then the larger this term

becomes and the greater the tendency to return to equilibrium. Taking equation (32) and expressing the left hand side of the equation as the material derivative and essentially integrating along characteristic of the equation we may obtain:

$$\frac{f(\xi,\underline{r}+\xi\delta_t,t+\delta_t)-f(\xi,\underline{r},t)}{\Delta t}=-\frac{1}{\tau}(f(\xi,\underline{r},t)-f^{(0)}(\rho,\underline{u})),$$
$$f(\xi,\underline{r}+\xi\delta_t,t+\delta_t)=f(\xi,\underline{r},t)-\frac{\Delta t}{\tau}(f(\xi,\underline{r},t)-f^{(0)}(\rho,\underline{u})). \qquad (33)$$

The essential idea of lattice Boltzmann is to solve this equation on a regular lattice made-up from a 'basis' or unit cell of discrete velocity vectors (see figure 1). This discrete velocity set c_i will then define all the positions on the lattice.

The D2Q9 LBE method operates on a two dimensional ($D = 2$) lattice with nine ($Q = 9$) possible velocity vectors. The lattice is essentially square with eight velocity vectors connecting nearest and next–nearest lattice nodes, the ninth, zero vector corresponds to the central rest node. The velocity vector direction is indexed by the quantity "i", the velocity vectors are denoted using $\underline{c}_i$ as shown (figure 1).

Each of the nodes depicted in figure 1 are arranged on a regular lattice that covers the simulation domain (figure 2).

The magnitude of the velocity vector is denoted as the vector population distribution function, f_i. The vector populations (f_i) are controlled by a set of governing laws that may be used to obtain the macroscopically observable quantities of the flow.

In General LB is a directional discretization of equation (33) in velovity space:

$$f_i(\mathbf{r}+\mathbf{c}_i\delta_t,t+\delta_t)\equiv f_i^{\dagger}(\mathbf{r},t)=f_i(\mathbf{r},t)+\Omega_{ij}\left(f_j^{(0)}(\rho,\mathbf{v})-f_j(\mathbf{r},t)\right)+\phi_i. \qquad (34)$$

In equation (34) Ω_{ij} is known as the collision matrix and f_i is the population of particles travelling with velocity $\underline{c}_i$ [3]. The term ϕ_i will be considered in greater detail in later sections, for current purposes it may be regarded as a term used to

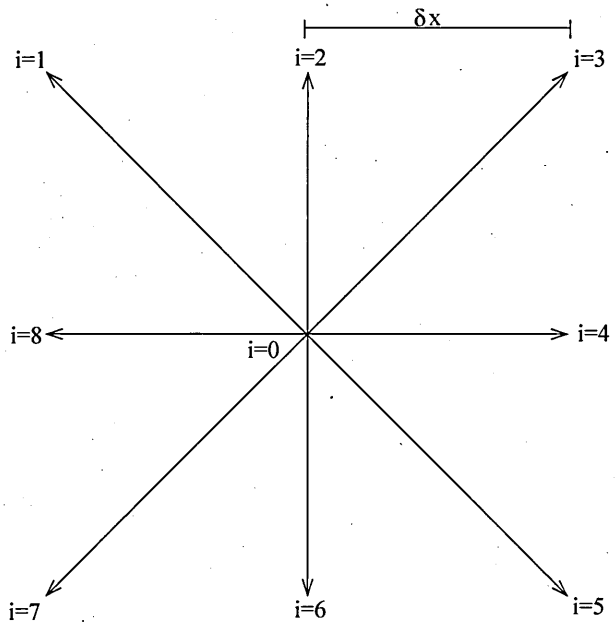

Figure 1: Single lattice node for D2Q9. The velocity vectors point in the directions indicated by the arrows. Note for i odd, velocity magnitude is increased by $\sqrt{2}$. Note that $|\underline{c}_4| = \frac{\delta_x}{\delta_t}$.

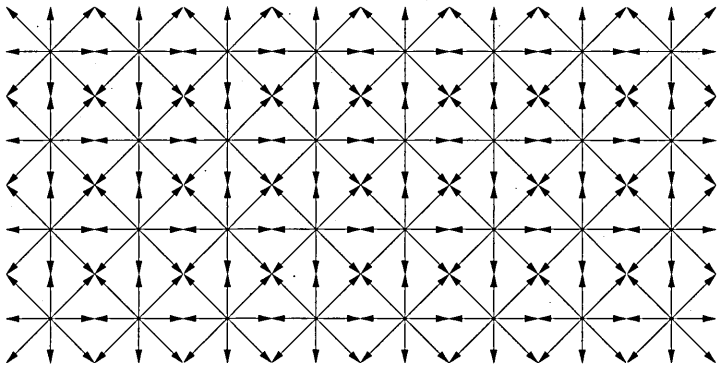

Figure 2: Lattice domain for D2Q9. The horizontal distance between adjacent nodes is the lattice spacing δ_x.

impose an external force on the fluid such as a gravitational field. In general, the lattice fluid's kinematic viscosity is determined by the principal eigenvalue of the symmetric collision matrix $\Omega_{ij} = \Omega_{ji}$. For the single relaxation time so-called LBGK model Ω_{ij} takes a simple form:

$$\Omega_{ij} = \frac{1}{\tau}\delta_{ij}, \tag{35}$$

with the collision parameter τ determining the fluid kinematic viscosity; for our particular D2Q9 choice [3]:

$$\nu = \left(\frac{2\tau - 1}{6}\right)\frac{\delta x^2}{\delta_t}, \tag{36}$$

where δ_x is the lattice spacing see figure 1 and δ_t is the time step. Note that the lattice spacing $\delta x = 1$ throughout. The Ω_{ij} or τ are assumed to be known.

For the D2Q9 model of Qian and d'Humieres [4], analysed in detail by Hou *et al.* [27];

$$f_i(\mathbf{r} + \mathbf{c}_i\delta_t, t + \delta_t) = f_i^{\dagger}(\mathbf{r}, t) = f_i(\mathbf{r}, t) + \frac{1}{\tau}(f_i^{(0)}(\rho, \mathbf{v}) - f_i(\mathbf{r}, t)) + t_p G, \tag{37}$$

$f_i^{\dagger}(\mathbf{r}, t)$ is the "post–collision", "pre–propagate" (see below) value of the momentum density; $t_p G$ is a "source" term, where t_p is the lattice link weight, which may be used for example, to represent a uniform pressure gradient [3, 28] or interfacial tension as an external force, in the case of multiple immiscible fluids [3]; all other symbols have their usual meaning [4, 27].

Algorithmically, equation (37) is interpreted as a collision step (right hand side of (37)) followed by a "streaming" or "propagation" step in which the calculated value of the right hand side is moved in the direction of $\underline{c}_i$ for an interval δ_t to reach the next lattice node.

i	0(rest)	1	2	3	4	5	6	7	8
t_p	4/9	1/9	1/36	1/9	1/36	1/9	1/36	1/9	1/36
c_{ix}	0	1	1	0	-1	-1	-1	0	1
c_{iy}	0	0	1	1	1	0	-1	-1	-1

Table 1: Definition of the D2Q9 simulation lattice link vectors, $\mathbf{c}_i$.

For all LB models the equilibrium distribution function $f_i^{(0)}$ is linear in ρ:

$$f_i^{(0)}(\rho, \mathbf{v}) = \rho P^{(2)}\left(\mathbf{v} \cdot \mathbf{c}_i\right) , \tag{38}$$

where the second-order polynomial $P^{(2)}(\mathbf{v}\cdot\mathbf{c}_i)$ depends upon the particular LB model being used [3].

The principal contribution to the distribution function is the equilibrium distribution function, which for LBGK, is defined as:

$$f_i^{(0)}(\rho, \mathbf{v}) = t_p \rho \left[1 + \frac{\mathbf{v} \cdot \mathbf{c}_i}{c_s^2} - \frac{v^2}{2c_s^2} + \frac{(\mathbf{v} \cdot \mathbf{c}_i)^2}{2c_s^4}\right] . \tag{39}$$

We derive the macroscopic fluid density and momentum as follows:

$$\begin{aligned} \rho &= \sum_i f_i, \\ \rho \mathbf{v} &= \sum_i f_i \mathbf{c}_i. \end{aligned} \tag{40}$$

It will be shown below how the form of equation (39) is arrived at.

In the present context equations (39) and (40) refer to those lattice links $\mathbf{c}_i$ and weights t_p defined in Table 1 and depicted in figure 1 where the speed of sound $c_s = 1/\sqrt{3}$ for the D2Q9 lattice. The above choice for the equilibrium distribution function also recovers a pressure tensor:

$$\Pi_{\alpha\beta}^{(0)} \equiv \frac{1}{3}\rho\delta_{\alpha\beta} + \rho v_\alpha v_\beta = \sum_i f_i^{(0)} c_{i\alpha} c_{i\beta} . \tag{41}$$

Note that the quantities ρ and v are the solutions to the Navier Stokes and

Continuity equations for the flow problem. It is not a trivial exercise to show that, as we shall now see.

4.1.2 Chapman Enskog Expansion for Macroscopic Dynamics

Within this section I attempt a limited but direct derivation of the hydrodynamic equations recovered from the Boltzmann equation via the Chapman Enskog expansion. Key points arising within this derivation will be referred to in following sections notably Chapter 6.

The Chapman-Enskog [29] analysis allows us to recover the hydrodynamic equations from the lattice Boltzmann evolution equations. The LB equation is initially expanded to second order accuracy in space and time with relation to the Knudsen number [3]. Through algebraic manipulation and scaled expansion a set of equations may be obtained at short and long distance scales and fast and slow time scales.

Working from the LBGK evolution equation we shall show how to arrive at the hydrodynamic equations and derive a system of simultaneous equations for parameters in $f_i^{(0)}$.

We begin with the LBGK evolution equation from equation (37) that is clearly space and time discretised:

$$f_i(\underline{x} + \delta_t c_i, t + \delta_t) = f_i(\underline{x}, t) + \frac{1}{\tau}(f_i{}^{(0)} - f_i(\underline{x}, t)). \tag{42}$$

Taylor expansion of equation (42) in two dimensions, retaining terms up to and including second order along with cancellation of like terms yields:

$$\delta_t(c_i.\nabla + \frac{\partial}{\partial t})f_i + \frac{\delta_t{}^2}{2!}(c_i.\nabla + \frac{\partial}{\partial t})^2 f_i = \frac{1}{\tau}(f_i^{(0)} - f_i(\underline{x}, t)). \tag{43}$$

The time and space derivatives are now Chapman Enskog expanded into parts consisting of their long and short components, this effective expansion of the variables

formally being carried out with relation to the Knudsen number, δ. We therefore make the following expansions:

$$\partial_t = \partial_{t0} + \delta\partial_{t1}, \tag{44}$$

$$f_i = f_i^{(0)} + \delta f_i^{(1)} + \delta^2 f_i^{(2)}. \tag{45}$$

Substituting the above definitions into equation (43) we obtain an expression that may subsequently be separated into first and second moments in terms of the Chapman Enskog Knudsen number expansion parameter δ which is set equal to δ_t:

$$\begin{aligned}
&\delta(c_i.\nabla + \partial_{t0} + \delta\partial_{t1})(f_i^{(0)} + \delta f_1^{(1)} + \delta^2 f_i^{(2)}) \\
&+\frac{\delta^2}{2!}(c_i\nabla + \partial_{t0} + \delta\partial_{t1})(c_i\nabla + \partial_{t0} + \delta\partial_{t1})(f_i^{(0)} + \delta f_i^{(1)} + \delta^2 f_i^{(2)}) \\
&= -\frac{1}{\tau}(\delta f_i^{(1)} + \delta^2 f_i^{(2)}).
\end{aligned} \tag{46}$$

Collecting and separately equating terms in δ and δ^2 it is possible to express the first and second moments of the distribution function as;

$$(c_i.\nabla + \partial_{t0}) f_i^{(0)} = -\frac{1}{\tau} f_i^{(1)}, \tag{47}$$

$$(c_i\nabla + \partial_{t0}) f_i^{(1)} + \partial_{t1} f_i^{(0)} + \frac{1}{2!}(c_i\nabla + \partial_{t0})^2 f_i^{(0)} = -\frac{1}{\tau} f_i^{(2)}. \tag{48}$$

Through straightforward algebraic manipulation of equation (48) and insertion of equation (47) we can obtain a form of equation (48) that is more easily manipulated within the analysis:

$$(1 - \frac{1}{2\tau})(c_i\nabla + \partial_{t0}) f_i^{(1)} + \partial_{t1} f_i^{(0)} = -\frac{1}{\tau} f_i^{(2)}. \tag{49}$$

We now have a set of equations that consider the first and second moments of the parameter δ namely equation (47) and equation (49) respectively. We now consider

the lattice basis summation moments that are imposed on the distribution function. For the conservation of mass, Chapmann and Enskog required:

$$\Sigma_i f_i^{(0)} = \rho,$$
$$\Sigma_i f_i^{(n>0)} = 0. \tag{50}$$

Similarly for momentum considerations we may follow Chapmann and Enskog:

$$\Sigma_i f_i^{(0)} \underline{c_i} = \rho \underline{u},$$
$$\Sigma_i f_i^{(n>0)} \underline{c_i} = \underline{0}. \tag{51}$$

Taking the equation for the first moment $\delta o(1)$, equation (47) and summing on "i" and inserting the above momentum and mass conservation considerations we can show that we have produced the continuity equation with short time scale components [30]. Expressed in tensor notation, we have;

$$\partial_{t0}\rho + \partial_\alpha \rho u_\alpha = 0. \tag{52}$$

Considering the second moment of $o(\delta)$, equation (47) we can write the equation in tensor notation and multiply through by $c_{i\beta}$:

$$\Sigma_i c_{i\alpha}\partial_\alpha f_i^{(0)} c_{i\beta} + \Sigma_i \partial_{t0} f_i^{(0)} c_{i\beta} = -\frac{1}{\tau}\Sigma_i f_i^{(1)} c_{i\beta}. \tag{53}$$

Using the identities expressed in equations (50) and (51), we may express equation (53) as;

$$\Sigma_i c_{i\alpha}\partial_\alpha f_i^{(0)} c_{i\beta} + \partial_{t0}\rho u_\beta = 0. \tag{54}$$

With an eye on the required emergent behaviour (see below) we require the "equilib-

rium" contribution $f_i^{(0)}$, of the distribution function f_i to have the following property;

$$\Sigma_i f_i^{(0)} c_{i\alpha} c_{i\beta} = c_s^2 \rho \delta_{\alpha\beta} + \rho u_\alpha u_\beta. \tag{55}$$

Substitution of (55) into (54) yields;

$$\partial_{t0} \rho u_\beta + \partial_\alpha c_s^2 \rho \delta_{\alpha\beta} + \partial_\alpha \rho u_\alpha u_\beta = 0. \tag{56}$$

Through tensor algebra we can exploit the property of the Kronecker Delta function in order to change the equation subscripts and subsequently in order to expand the equation further we invoke the product rule for differentiation. Cancellation of terms may then be carried out using the continuity equation, equation (52), to transform terms within the equation. The equation resulting from these operations is:

$$\partial_{t0} u_\beta + u_\alpha \partial_\alpha u_\beta = -\partial_\beta c_s^2 ln(\rho). \tag{57}$$

Comparison of equation (57) with earlier work shows that our model reproduces a form of the Euler equation on short time scales. Thus far we have used the equation containing terms of first order in "δ" to recover the continuity and Euler equations on short time scales.

We can now proceed to considerations of the longer time scale $o(\delta^2)$, equation (49). Summation of equation (49) on "i" and taking into account the mass and momentum conservation constraints identified in equations (50) and (51) we have the equation;

$$\partial_{t1} \rho = 0. \tag{58}$$

Multiplication of equation (58) with "δ" and adding with, from the previous analysis, the short time scale continuity equation, equation (52) we obtain, through standard factorization, and consideration of equation (44), a demonstration that our model

recovers the continuity equation on long time scales:

$$\partial_t \rho + \partial_\alpha (\rho \underline{u_\alpha}) = 0. \tag{59}$$

In order to obtain further dynamics of the system we return to the $o(\delta^2)$ equation, equation (49). Multiplication throughout by a second velocity vector $c_{i\beta}$ prior to summation on the "i" components yields the following;

$$\partial_{t1} \Sigma_i f_i^{(0)} c_{i\beta} + (1 - \frac{1}{2\tau}) \Sigma_i (\partial_{t0} + c_{i\alpha} \partial_\alpha) f_i^{(1)} c_{i\beta} = -\frac{1}{\tau} \Sigma_i f_i^{(2)} c_{i\beta}, \tag{60}$$

to which we return to in Chapter 6, note.

Using equation (51), equation (60) may be simplified (having made a substitution for the first moment of the distribution function, f_i, in conjunction with equation (47) in tensor format):

$$\partial_{t1} \rho u_\beta + (1 - \frac{1}{2\tau}) \partial_\alpha \Sigma_i (-\tau (\partial_{t0} + c_{i\gamma} \partial_\gamma) f_i^{(0)}) c_{i\alpha} c_{i\beta} = 0. \tag{61}$$

Through general expansion and substitution of the definitions made in equation (55) we obtain from the above a form that may be expanded further using the product rule for differentiation:

$$\partial_{t1} \rho u_\beta + (\frac{1}{2} - \tau) \partial_\alpha \left[\partial_{t0} c_s^2 \rho \delta_{\alpha\beta} + u_\beta \partial_{t0} \rho u_\alpha + \rho u_\alpha \partial_{t0} u_\beta + \partial_\gamma \Sigma_i f_i^{(0)} c_{i\gamma} c_{i\alpha} c_{i\beta} \right] = 0. \tag{62}$$

The second and third terms contained within the square bracket in equation (62) may be expanded using definitions derived from the Euler equation on a short time scale, equation (57), giving;

$$\begin{aligned} &\partial_{t1} \rho u_\beta + (\frac{1}{2} - \tau) \partial_\alpha [\partial_\gamma \Sigma_i f_i^{(0)} c_{i\gamma} c_{i\alpha} c_{i\beta} + \partial_{t0} c_s^2 \rho \delta_{\alpha\beta} \\ &- u_\beta c_s^2 \partial_\alpha \rho - u_\beta \partial_\gamma \rho u_\gamma u_\alpha - u_\alpha c_s^2 \partial_\beta \rho - u_\alpha \rho u_\gamma \partial_\gamma u_\beta] = 0. \end{aligned} \tag{63}$$

The fourth and final terms within the square bracket may now be combined using the reverse of the product rule for differentiation resulting in;

$$\partial_{t1}\rho u_\beta + (\frac{1}{2} - \tau)\partial_\alpha[\partial_\gamma \Sigma_i f_i^{(0)} c_{i\gamma} c_{i\alpha} c_{i\beta} + \partial_{t0} c_s^2 \rho \delta_{\alpha\beta}$$
$$-u_\beta c_s^2 \partial_\alpha \rho - u_\alpha c_s^2 \partial_\beta \rho - \partial_\gamma \rho u_\alpha u_\beta u_\gamma] = 0. \quad (64)$$

The final term in the square bracket of equation (64) is conventionally called the error term and is an undesirable but unavoidable occurrence within the analysis. Dropping the error term from the analysis and substituting in the continuity equation in order to remove the time derivative from the second term within the bracket, results in;

$$\partial_{t1}\rho u_\beta + (\frac{1}{2} - \tau)\partial_\alpha[\partial_\gamma \Sigma_i f_i^{(0)} c_{i\gamma} c_{i\alpha} c_{i\beta} - c_s^2 \partial_\alpha \rho u_\alpha \delta_{\alpha\beta}$$
$$-u_\beta c_s^2 \partial_\alpha \rho - u_\alpha c_s^2 \partial_\beta \rho] = 0. \quad (65)$$

The time derivative $\partial_{t1}\rho u_\beta$ will be incorporated naturally into the overall time derivative when equations (57)and (65) are combined it is disregarded for the moment. By considering only the function contained within the square brackets we obtain a further constraint on the equilibrium distribution function required to achieve Navier-Stokes behaviour:

$$\partial_\gamma \Sigma_i f_i^{(0)} c_{i\gamma} c_{i\alpha} c_{i\beta} - c_s^2 \partial_\alpha \rho u_\alpha \delta_{\alpha\beta} - u_\beta c_s^2 \partial_\alpha \rho - u_\alpha c_s^2 \partial_\beta \rho = k\rho S_{\alpha\beta} \quad (66)$$

Henceforth we shall also neglect the term that is quadratic in velocity. Through calculus and general algebraic manipulation the above expression may be manipulated into a form resembling the Navier-Stokes equation [30].

In order to recover the Navier-Stokes behaviour, the above equation must be equal to a constant multiplied by the density and stress as shown. We now consider the general form of the equilibrium distribution function that must be used to make

certain that the steps above may be achieved. We write:

$$f_{i\neq 0}^{(0)} \approx A_\sigma + B_\sigma u_\delta c_{i\delta} + C_\sigma u_\delta u_t c_{i\delta} c_{it} + D_\sigma u_\delta u_\delta,$$
$$f_{i=0}^{(0)} \approx A_\sigma + D_\sigma u_\delta u_\delta. \tag{67}$$

By recalling the constraints that have been applied to the equilibrium distribution function in order to recover the hydrodynamic equations in the above analysis it is possible to generate a system of simultaneous equations in A...D that allow us to determine values for the constants in the general form of the equilibrium distribution function equation. For convenience the constraints on the equilibrium distribution function we have just used are summarized again below:

$$\Sigma_i f_i^{(0)} = \rho,$$
$$\Sigma_i f_i^{(0)} c_{i\alpha} = \rho u_\alpha,$$
$$\Sigma_i f_i^{(0)} c_{i\alpha} c_{i\beta} = c_s^2 \rho \delta_{\alpha\beta} + \rho u_\alpha u_\beta,$$
$$\partial_\gamma \Sigma_i f_i^{(0)} c_{i\alpha} c_{i\beta} c_{i\gamma} = c_s^2 (\delta_{\alpha\beta} \partial_\gamma \rho u_\gamma + u_\beta \partial_\alpha \rho + u_\alpha \partial_\beta \rho) + k \rho S_{\alpha\beta}. \tag{68}$$

For the lattice depicted in figure 1 (and indeed any viable LB simulation lattice) it may be shown that:

$$\sum_i t_p c_{i\alpha} = 0,$$
$$\sum_i t_p c_{i\alpha} c_{i\beta} = k_2 \delta_{\alpha\beta},$$
$$\sum_i t_p c_{i\alpha} c_{i\beta} c_{i\gamma} = 0,$$
$$\sum_i t_p c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} = k_4 (\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}), \tag{69}$$

where $k_2 = \frac{1}{3}$, and $k_4 = \frac{1}{9}$ (for the D2Q9 lattice).

Essentially the first three of the above constraints will determine the form of the equilibrium distribution function and the value of the models speed of sound c_s. The fourth of the constraints uses the speed of sound to identify exactly the value of the constant k in equation (68) as $\frac{1}{6}$. Equilibrium distribution functions summations in (68) are now decomposed into separate summations of the rest link and of the short links (+) and of the long links (×). Considering the first of the constraints summarized in system (68);

$$\Sigma_i f_i^{(0)} = A_0 + 4A_1 + 4A_2 + B_1\Sigma^+ u_\delta c_{i\delta} + B_2\Sigma^\times u_\delta c_{i\delta} + C_1\Sigma^+ u_\delta u_t c_{i\delta} c_{it},$$
$$+C_2\Sigma^\times u_\delta u_t c_{i\delta} c_{it} + (D_0 + 4D_1 + 4D_2)u^2 = \rho. \tag{70}$$

The above condition must hold for all values of fluid velocity, $\underline{v}$. We therefore obtain two equations governing the coefficients of the equilibrium distribution function:

$$A_0 + 4A_1 + 4A_2 = \rho, \tag{71}$$

$$2C_1 + 4C_2 + D_0 + 4D_1 + 4D_2 = 0. \tag{72}$$

The second of the equilibrium distribution function weighted summations in (68) may now be considered;

$$\Sigma_i f_i^{(0)} c_{i\alpha} = B_1\Sigma^+ u_\delta c_{i\delta} c_{i\alpha} + B_2\Sigma^\times u_\delta c_{i\delta} c_{i\alpha}$$
$$+C_1\Sigma^+ u_\delta u_t c_{i\delta} c_{it} c_{i\alpha} + C_2\Sigma^\times u_\delta u_t c_{i\delta} c_{it} c_{i\alpha}$$
$$+D_0 c_{i\alpha} u^2 + 4D_1 c_{i\alpha} u^2 + c_{i\alpha} 4D_2 u^2 = \rho u_\alpha. \tag{73}$$

Using lattice moment properties defined above we remove all terms not quadratic in $c_{i\alpha}$. Cancellation of terms results in the following restraint on the coefficients of

the equilibrium distribution function;

$$2B_1 + 4B_2 = \rho. \tag{74}$$

Performing similar operations with the second and third velocity moments allows us to produce further relations between the coefficients in the equilibrium distribution function defined in (70). When considering the third c_i moment we force a condition on the C coefficients, namely;

$$2C_1 - 8C_2 = 0. \tag{75}$$

This condition insures Gallilean invariance [31]. Making this assumption within the second c_i moment of the equilibrium distribution function allows us to make the following statements:

$$2(A_1 + 2A_2) = c_s^2\rho, \tag{76}$$

$$8C_2 = \rho, \tag{77}$$

$$4C_2 + 2(D_1 + 2D_2) = 0. \tag{78}$$

We can now consider the third c_i moment of the equilibrium distribution function in equation (68) which generates the following constraint on the coefficients of the equilibrium distribution function:

$$2B_1 - 8B_2 = 0. \tag{79}$$

Manipulating the third velocity moment into a form to generate the Navier-Stokes equation gives a final restriction on the equilibrium distribution function [31]:

$$B_2 = \frac{c_s^2\rho}{4}. \tag{80}$$

We have now produced a set of simultaneous equations that restrict the coefficients of the equilibrium distribution function in such a way as to allow us to recover conventional hydrodynamics namely the continuity and Navier-Stokes equations from the lattice Boltzmann equation algorithm. The system of equations however is not a conventional system of equations, as it is under-determined i.e. we have a system of eight equations in ten unknowns. The system also contains subsets of equations, these subsystems may be solved exactly:

$$
\begin{aligned}
A_0 + 4A_1 + 4A_2 &= \rho, \\
2A_1 + 4A_2 &= c_s^2 \rho, \\
B_1 + 4B_2 &= \rho, \\
2C_1 + 4C_2 + D_0 + 4D_1 + 4D_2 &= 0, \\
2C_1 - 8C_2 &= 0, \\
8C_2 &= \rho, \\
4C_2 + 2D_1 + 4D_2 &= 0, \\
2B_1 - 8B_2 &= 0.
\end{aligned}
\tag{81}
$$

The subsystems contained within the set of eight equations allows the coefficients B_1, B_2, C_1 and C_2 to be solved completely. It is also worth noting that the B_2 coefficient determines the speed of sound c_s^2 in the calculations. Through equation (76) the value of B_2 also constrains the values of the coefficients A_1 and A_2 therefore the range of "free" coefficients is somewhat restricted. We can note that the values selected for the "free variables" will effect the overall properties of the system. The choice of A_0 for instance will determine the amount of rest mass contained on a link. The selection of a large value of A_0 will restrict the attainable Mach number of the flow as there is less momentum to be distributed on the non-zero velocity links,

$i = 1 - 8$. Having solved for A...D, the $f_i^{(0)}$ in LBGK may be written:

$$f_i^{(0)}(\rho, \mathbf{v}) = t_p \rho \left[1 + \frac{\mathbf{v} \cdot \mathbf{c}_i}{c_s^2} - \frac{v^2}{2c_s^2} + \frac{(\mathbf{v} \cdot \mathbf{c}_i)^2}{2c_s^4} \right] . \tag{82}$$

4.1.3 Length and Time Scale Considerations

When considering the behaviour of momentum transfer within a fluid, as would be expected, internal friction causes the fluid distribution function to return to thermodynamic equilibrium. The momentum of the particles of the fluid and thus the information that they carry is transferred through collisions. Low density systems such as low density gases transfer information slowly as collisions between particles are rare, the length scales, viscosity and density of the simulation therefore influence the rate at which a simulation achieves its equilibrium velocity distribution. The equilibrium state of a fluid may be achieved through different processes depending on the time scales involved.

A first relaxation regime occurs when the system under consideration consists of many bodies and the time scales involved are shorter than the duration of a collision event. Due to the short time scales there must be rapid relaxation of the many body distribution function, occurring at the collision, to the single particle distribution function. In the Boltzmann regime the collisions are assumed to be instantaneous and therefore the relaxation mechanism considered in this system are ignored.

The second relaxation mode is the kinematic regime where the time scales are in the region of the duration of the time of flight between collision events. The Maxwellian equilibrium distribution is a many body ensemble average over a local region where the collision occurred. The system described in this regime is a series of local regions in local Maxwellian equilibrium.

The third possible relaxation mode is the hydrodynamic regime and is achieved when the relaxation time scales lie between the time elapsed between particle collisions and the minimum hydrodynamic time scales considered. The equilibrium

is once again a Maxwellian distribution however unlike the kinematic regime, the hydrodynamic relaxation occurs as a slow drift to a global Maxwellian. The time scales involved in this mode of thermodynamic equilibration are too long to consider alongside (i.e. in the same calculation) those previously discussed. These long time scales, may only be resolved with current and all anticipated computational resources in the absence of the more transient phenomena. Long time scales are central as they characterise the flow processes of interest. Essentially, as the systems under consideration here are continually changing (by flowing), so local distributions attempt to stabilize and form a global equilibrium. However, the constant injection of energy (say) makes this equilibration very slow; the systems (attempted) approach to equilibrium is the process of interest.

4.1.4 Stress in LBGK Simulations

Recalling equation (60) we can identify a key practical advantage of the LB technique that is that the stress, $\sigma_{\alpha\beta}$ and the viscous stress $\sigma'_{\alpha\beta}$ (see section 3.2.3 for definitions) are carried in the f_i distribution locally which shall now be shown. The local information may be used when e.g. considering Lagrangian particles or interfaces between two immiscible fluids.

We can make use of the conservation conditions imposed on the distribution function of equation (68) and recall the summation conventions of equation (51) to modify equation (60) giving;

$$\partial_{t1}\rho u_\beta + (1 - \frac{1}{2\tau})\partial_\alpha \Sigma_i f_i^{(1)} c_{i\alpha} c_{i\beta} = 0, \tag{83}$$

which when added to the $o(\delta)$ expression (57) gives the Navier-Stokes equation. The following identity may therefore be made for the viscous stress:

$$\sigma'_{\alpha\beta} = (1 - \frac{1}{2\tau})\Sigma_i f_i^{(1)} c_{i\alpha} c_{i\beta}. \tag{84}$$

Apparently the viscous stress may be derived directly from local distribution functions, $f_i^{(1)}$. This is a considerable advantage relative to other numerical methods. It means that the fluid stress may be constructed from local information, instead of from measured velocity gradients (which must be measured from non–local finite differences).

From the alternate form of the evolution equation (37), in Chapter 8 we will make use of the concept of pre and post-collision quantities. Equation (37) may, with some of the concepts established in the previous section, be written as;

$$f_i^{\dagger} = f_i + \frac{1}{\tau}(f_i^{(0)} - f_i), \tag{85}$$

a dagger denotes a post collision quantity, whence;

$$f_i^{\dagger} = f_i^{(0)}(\rho, \underline{u}) + f_i^{(1)} + f_i^{(2)} - \frac{1}{\tau}(f_i^{(1)} + f_i^{(2)} + ...). \tag{86}$$

Therefore the post-collision f_i is given by;

$$f_i^{\dagger} \approx f_i^{(0)} + \left(1 - \frac{1}{\tau}\right) f_i^{(1)}. \tag{87}$$

Recalling the relationship between $f_i^{(1)}$ and the stress (equation (84)) LB methods may be styled "stress-relaxation" methods and;

$$f_i^{(1)\dagger} = \left(1 - \frac{1}{\tau}\right) f_i^{(1)}, \tag{88}$$

with $f_i^{(0)}$ being considered in the collision process of equation (87).

4.2 Multi–component LB Methods

4.2.1 Introduction to Multiphase LB Methods

The work carried out so far considers single phase fluid flows only, we shall now consider the behaviour of mutiphase flows and the relevant extensions required to recover it. The behaviour of multiphase flows will be briefly outlined along with a multi–component LB component segregation method (the essential extra ingredient) that has been constructed and is the subject of later chapters of this thesis.

Multiphase behaviour is recovered by considering a single component continuum fluid with unstructured interfaces (where the structure of the interface is not of concern, the interface is considered a transition from one fluid to another). Incorporated into the behaviour of the single continuum fluid are two conditions, namely; a body force giving rise to interfacial tension and a coupled phase field which tells the body force where to act. When combined, the three components of the single continuum fluid, the body force and the phase field give rise to a multi–component system, which in this work are considered to be "red" and "blue" fluids.

As we have already considered single phase fluid behaviour (that governs the behaviour of the underlying continuum fluid) earlier in this thesis I shall now look at the remaining algorithmic components required to achieve a multiphase flow.

4.2.2 Multiphase Extensions and Definitions

This section contains a formal but brief introduction to multi–component generalisations; a less formal and more formulaic and practical account is contained later in section 7.4 where multi–component methods are more fully considered.

In calculation the same basic mass and momentum conservation rules still apply however we make the following generalization;

$$f_i \rightarrow (R_i, B_i), \tag{89}$$

$$\rho \rightarrow (R, B). \tag{90}$$

In order to distinguish between immiscible lattice fluids, the multi–component LBE uses a phase field based upon the densities of "red" and "blue" fluids present at a node.

Firstly we shall consider the phase field. The phase field is characterized by ρ^N with a range of ± 1, with $+1$ representing completely red fluid and -1 representing completely blue fluid. Following the notation of [25], we define a component index, or phase field, $\rho^N(\mathbf{r})$:

$$\rho^N(\mathbf{r}, t) \equiv \left(\frac{\rho_R(\mathbf{r}, t) - \rho_B(\mathbf{r}, t)}{\rho_R(\mathbf{r}, t) + \rho_B(\mathbf{r}, t)}, \right) \tag{91}$$

where $-1 \leq \rho^N(\mathbf{r}) \leq 1$. Note that ρ^N varies in time only through time dependence in $\rho_R(\mathbf{r}, t)$ and $\rho_B(\mathbf{r}, t)$. Red and blue fluids mix under the LBE propagation step [3]; this defines the interfacial region, i.e. the transition from red to blue fluid.

We can now define a vector known as the colour field, the colour field is a unit vector that lies normal to the interface between the red and blue fluids. The vector direction is so that it points from the blue phase into the red phase:

$$\underline{\hat{f}_\sigma} = \frac{\nabla \rho^N}{|\nabla \rho^N|}. \tag{92}$$

For the above calculation it can be shown that the gradient of the phase field can be obtained from an "isotropic derivative" see section 7.5.3;

$$\nabla \rho^N = 3 \Sigma_i t_p \rho^N (r + c_i) c_i. \tag{93}$$

The curvature of the interface may be calculated using the colour field unit vector,

in tensor notation the curvature of the interface is given by (see [32]);

$$k = (\delta_{\alpha\beta} - f_{\sigma\alpha} f_{\sigma\beta}) \partial_\beta f_\alpha. \tag{94}$$

It is worth noting at this stage that the calculation of curvature outlined in equation (94) is dependant on the gradient of the colour field and thus assumes that the interface is continuous. This issue shall be revisited in the final chapter, Chapter 11, when the interface at a boundary is no longer continuous.

Now that we have derived the main substitutions and primary quantities, collision is carried out as before for a single continuum fluid with f_i given by $R_i + B_i$. Post collision, the populations are re-allocated so as to promote segregation between the two fluids with $R = \Sigma_i R_i$ and $B = \Sigma_i R_i$. The method of segregation determines the properties and dynamics of the phase field.

4.3 Interface Implementation

The several multi–component LB methods may be distinguished by the different ways in which they impose a fluid-fluid interface. Illustrative references are [21, 22, 24, 25, 33]. See also references [3, 34] for a survey of the methods' relative advantages and applications. In problems where the kinematics of phase separation feature, Swift's method [21, 22], based as it is upon the Cahn–Hilliard theory, represents an appropriate choice of LBE interface algorithm. Here, however, we aim to address only completely immiscible mixtures in a continuum approximation, in which interfaces are assumed to be very narrow and unstructured. Computational resources or physical accuracy may also impose similar requirements and restrict choice of interface algorithm to a type pioneered by Gunstensen et al [24] and later modified by Lishchuk et al [25]. The model presented and applied here is based on the latter.

In Lishchuk's method (see section 4.3.3) the interface is based on the contin-

uum level stress boundary conditions which apply on an unstructured interface between completely separated fluids [5]; imposed through a body force acting in the fluid; an idea first introduced in the context of interfacial tension by Brackbill, Kothe and Zemach in 1992 [35]. Lishchuk's method produces narrow interfaces with reduced micro-currents (or 'spurious velocities' [3]), it has an independently adjustable interfacial tension and it sustains interfacial tensions larger than Gunstensen's method [24].

4.3.1 Numerical Segregation or Re–colour

The method referred to in this work as numerical re–colour was suggested by Gunstensen and Rothman [24]. The re–colour method is essentially a maximisation problem. The lattice fluid is initially collided in a colourblind collision. Post collision, the lattice densities are subjected to a perturbation that has the effect of inducing a surface tension before the re–colour step and subsequent propagation.

The first step of the numerical re–colour algorithm is to identify the interface region from the magnitude of the phase field vector. Note, the phase field vector also lies perpendicular to the interface direction. In regions where the gradient of the phase field implies that there is an interface between two binary fluids, a perturbation is added to the collided densities. The effect of this perturbation is to redistribute the link densities, f_i, within the interface region; diminishing populations on links parallel to the interface and increasing populations perpendicular to the interface subject to the constraints of mass conservation.

In practice, the application of the numerical re–colour algorithm is carried out using the direction of the colour field unit vector which is used to generate "filling priority" tables for the nine velocity links of the D2Q9 lattice system. Links with the largest component of their velocity lying in the direction of the colour field are assigned highest filling priority subsequent links are similarly assigned filling priority depending on the magnitude of their component parallel to the colour field

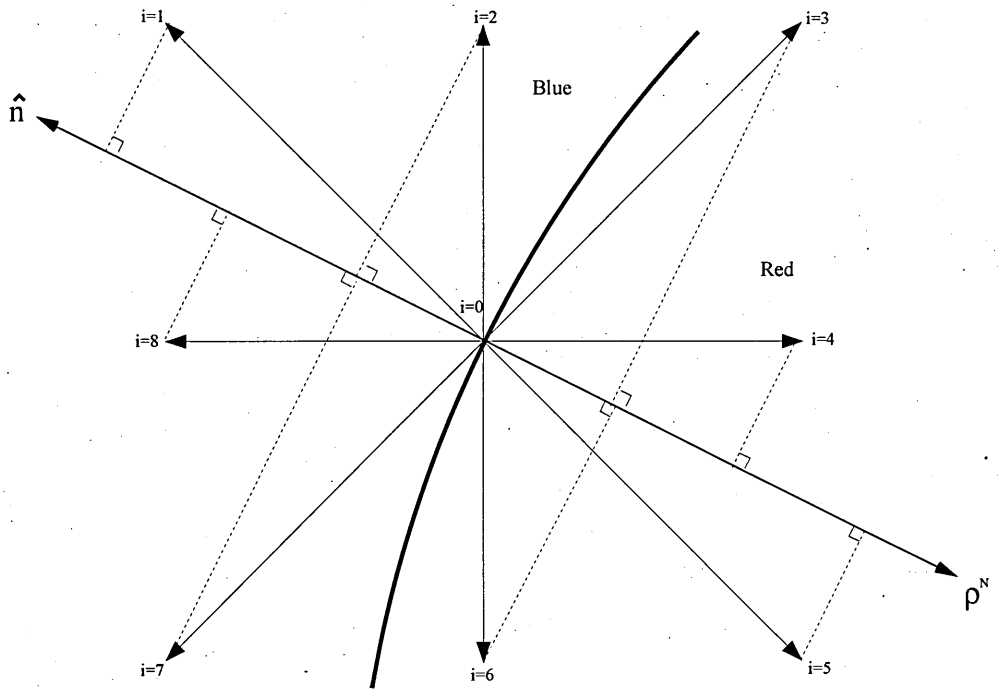

Figure 3: Single interface lattice node for D2Q9 showing numerical re–colour filling priority

direction. In figure 3 for example, the interface is shown crossing a lattice node, the links of highest filling priority in terms of the red fluid are links $i = 5, 4, 6, 3, 7, 2, 8, 1$. Segregation is achieved, for the numerical method, by taking all of the red fluid lying at a lattice site and allocating as much as possible to the link of highest priority, links are filled according to their priority until there is no red fluid remaining on the site. The remaining, empty or partially filled links are then populated with the remaining blue fluid thus adhering to the conservation of mass at the node. The benefit of the numerical method of segregation is that it allows us to insert a surface tension between immiscible fluids and produces a relatively sharp interface, usually distributed over a maximum of three sights. The disadvantages of such a segregation method is the large amounts of noise that is generated at low Reynolds number flows. The method is also computationally labour intensive and slow. The development of the priority look up tables takes time but the subsequent repeated reallocation of colour onto the links is a massively time consuming process. The noise generated in

hydrodynamic quantities, other then the phase field such as velocity, when using this segregation method remains the most significant issue and undermines the advantage of the sharp interface, see Chapter 8 for more details.

4.3.2 Formulaic Segregation and the Diffuse interface

Working from a method proposed by Latva-Kokko and Rothman [6], segregation may be achieved using an algorithm that allows a degree of colour diffusion at the interface. The algorithm is applied post collision and gives segregated or re–coloured densities as;

$$R_i^{\dagger\dagger} = \frac{R}{\rho} f_i^{\dagger} + t_p \frac{RB}{\rho} \beta cos(\theta_i - \theta_f) \underline{c}_i, \tag{95}$$

where θ_i is the link angle, θ_f is the colour field angle. Here, the link and the colour field are both measured by the angle subtended at the local x-axis, taken positive in the anti–clockwise direction. β determines the extent of the diffusion of the red and blue fluids into one another. Here $R_i^{\dagger\dagger}$ is the red density allocated to the link "i", of course $B_i^{\dagger\dagger} = f_i - R_i^{\dagger\dagger}$. As β increases the interface becomes less diffuse. Latva-Kokko and Rothman state in [6] that the value of beta lies in the range $0 < \beta < 1$ however from simulation data we have found that the maximum value of β that can be used is in the region of 0.71, see equation (255). Simulations will run at higher values however situations such as negative densities have been found to occur, this point is discussed in greater detail in Chapter 8.

4.3.3 Interface Force; Lishchuk's Method

It is possible to show that the effect of interfacial tension existing between two immiscible fluids may be approximated using a body force. The magnitude of the body force will be dependent on the curvature of the fluid-fluid interface. We therefore suppose that the hydrodynamics in the region of the interface are governed by the continuity and Navier-stokes equations with an additional body force, $\underline{F}$. $\underline{F}$ is

defined in terms of the surface tension parameter, α as follows;

$$\underline{F} = \frac{\alpha}{2} k \nabla \rho^N, \tag{96}$$

where k is the curvature, identified above, of the interface which for low Reynolds number flows we can assume varies slowly and is in some cases constant i.e. spherical drops. The effect of the body force is to produce a pressure step across the interface ΔP;

$$\Delta P = \frac{\alpha k}{2} \int_{Blue}^{Red} \underline{F}.dl \approx \int_{Blue}^{Red} \frac{\alpha k}{2} \frac{\partial \rho^N}{\partial x} dx. \tag{97}$$

Through standard algebraic manipulations it can be shown that the pressure step at the interface may be expressed as;

$$\Delta P = \frac{\alpha k}{2} [\rho^N]_{-1}^{+1} = \alpha k, \tag{98}$$

as required to recover the Laplace law [25].

It is appropriate to note at this stage that the use of the curvature in the calculation of the interface inducing body force creates specific problems when the interface is intersected by a boundary in problems such as wetting. This issue is further discussed in Chapter 11 and illustrated by the cartoon in figure 4. Problems arise in the circled regions.

4.4 Navier Slip Condition

The Navier slip condition is primarily used in the latter stages of this work specifically in chapter 10, however it is deemed appropriate to introduce the concept at this point.

Consider an isolated droplet in contact with a boundary contained within a continuum fluid. As illustrated in figure 4. When a force, or gravity, is applied in the positive x-direction, the contact point between the two fluids and the boundary

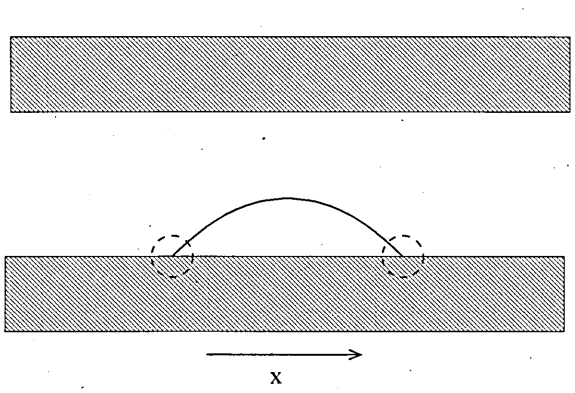

Figure 4: Isolated droplet in contact with channel boundary.

will move. The contact is then known as the dynamic contact line or DCL.

If a no-slip condition were applied to the boundary there would result a singularity at the contact point. In order to avoid the occurrence of such a singularity, a slip boundary condition may be implemented. There are a number of slip conditions that may be applied to the boundary however, in this work, the Navier slip condition is to be utilised. For alternative methods of applying a slip condition at the boundary the interested reader is directed to Happel and Brenner [12]. It is not the intention of this work to verify or to modify the Navier slip condition in any way, the condition is employed widely in calculation as a method of allowing slip velocity on the boundary to evolve and to avoid generating the aforementioned singularity. It is also worth stressing that this work does not aim at an algorithm that predicts the slip velocity (as with many algorithmic non-equilibrium molecular dynamic simulations currently under development).

The Navier slip condition we use here contains a parameter known as the slip length "b" this is taken to be a molecular property of the material itself. The Navier condition allows a slip velocity to evolve according to the following relationship when applied to the geometry of figure 5:

$$u_x(y=0) = b\frac{\partial u_x}{\partial y}. \tag{99}$$

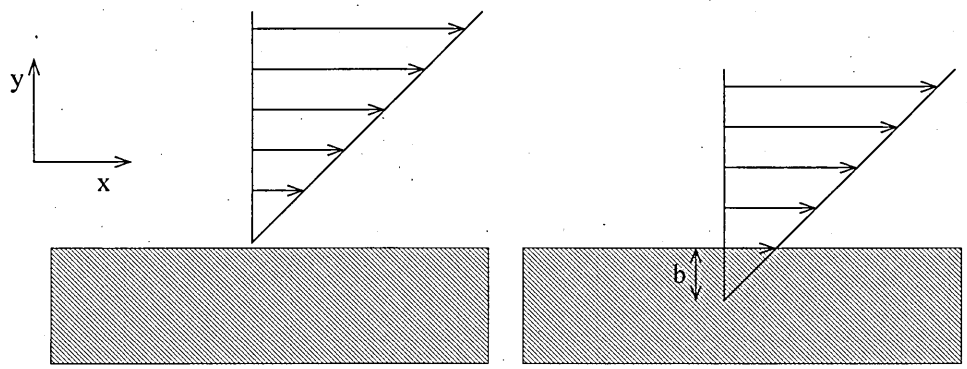

Figure 5: X-velocity profile as a representation of the Navier slip condition where b is the slip lenght

The role of the parameter "b", the slip length, may be envisaged as shown in figure 5.

4.5 Boundary Closure Methods

Lattice closure methods in LBE [3] comprise two sets. One provides Lees-Edwards type boundary conditions [36], typically for calculation of colloidal phase properties and phase separation kinematics [37]. The methods developed in this thesis belongs to a second class of LBE boundary closure, designed to impose a specified velocity distribution (possibly corresponding to a slip) on the defined surfaces which comprise the boundary.

There are several implementations of the rest boundary condition in LB [3]. The popular, convenient and reasonably accurate bounce–back method has been evaluated, at steady–state, by a number of workers (e.g. [38, 39]) and He and Zou *et al.* have shown how it is possible, analytically, to predict small, defect boundary velocities resulting from its use [40]. More recent, more accurate LB boundary algorithms apply to moving boundaries and involve bouncing–back the non-equilibrium part of the LB momentum densities [31]; extrapolation [41]; the introduction of a counter–slip velocity [42]; second order bounce–back [43] and Skordos [44] has solved the problem of boundary closure by deriving auxiliary partial differential

equations which may be solved for lattice boundary information. Of some importance here is one of the most overlooked and most general approaches to simulation lattice closure - the locally second-order boundary method (LSOB) of Ginzbourg and d'Humières [45].

Boundary closure plays a central role in this work. The next chapters consider the basic method I developed in the context of single component LB.

5 Mass Conserving Boundary Closure Scheme

In this chapter and the next I intend to deal with the fundamental issues surrounding boundary conditions associated with the single phase LB simulation. In order to maintain tractability, the analysis is to be carried out using single component lattice Boltzmann (SCLB). The work presented here will not be generalised to MCLB until Chapter 10, by which point the core multi–component lattice Boltzmann (MCLB) method will have been considered in detail in chapters 7 – 9 inclusive. MCLB therefore will not be used within this chapter or the next.

In addition to restricting our treatment here to single component fluids, we shall also suppose that the fluid is not subject to any external body force such as gravity imposed with the term ϕ_i in equation (34). In the next chapter we shall revise the last assumption.

5.1 Introduction

It is important to recall that accurate representation of boundary conditions is essential in all computational fluid dynamics (CFD). Here we derive, then demonstrate a robust simulation lattice closure algorithm for the single phase lattice Boltzmann (LB) [3] method. Our method treats the fluid on the boundary with the same accuracy as the bulk LB fluid and extends previous work carried out in [46] (i) by shifting emphasis away from boundary pressure onto boundary mass conservation (an adjustment made necessary by multi–component applications) and (ii) by applying the method to an internal and an external corner geometry with no diminution of accuracy. In these respects, the new method presented here is as convenient as the popular LB strategy of mid–link bounce–back [3]; however it has additional advantages of (i) applying boundaries on a given lattice node (not at the mid point between nodes), (ii) being instantaneously accurate (our lattice closure strategy is verifiably accurate after a very small number of simulation time steps), (iii) being

capable of representing moving fluid boundaries, including boundaries crossed by fluid and (iv) working with a range of collision parameters, τ.

Solutions of the Navier Stokes and continuity equations, $(\mathbf{v}(\mathbf{r},t), P(\mathbf{r},t))$, are most conveniently closed by Dirichlet boundary conditions on fluid velocity (only):

$$\mathbf{v}(\mathbf{r},t)|_{\mathbf{r}'t} = \mathbf{u}_0(\mathbf{r}',t), \tag{100}$$

$\mathbf{r}'$ denotes a position on the boundary of the flow domain; $\mathbf{u}_0(\mathbf{r}',t)$ denotes a specified boundary velocity. The several lattice closure methods in LB [3] reduce to imposition of a given $\mathbf{u}_0(\mathbf{r}',t)$. Note that $\mathbf{u}_0(\underline{r}',t)$ is supposed to be known throughout all that follows in this chapter and chapter 10.

In this chapter, we derive, test–bench and compare a portable *closure* step for application over the edges of a lattice Boltzmann fluid simulation space. The step is equivalent to applying Dirichlet boundary conditions. Accordingly, it assumes (i) a boundary velocity and (ii) local mass conservation. It (i) gives correct behaviour at every time-step; (ii) represents boundary lattice fluid to the same accuracy as the bulk lattice fluid and (iii) applies to immiscible fluids in contact at a boundary (see Chapter 10). The method is derived for a locally flat wall at which fluid is assumed to be entrained with a specified velocity; then show equivalent results for internal and external corners. To test the method we present (i) accuracy plots based upon exactly solved flows and (ii) stream functions derived from the lid driven cavity.

It is henceforth assumed that our LB boundary node lies infinitesimally *within* the lattice fluid, which, locally, is moving at specified velocity. We work in 2D, using the popular D2Q9 lattice, as described earlier. However, the analysis generalizes to 3D directly. For consistency and simplicity, as much of the notation and approach of the extended work of [46] is used as possible. Our purpose then: momentum densities, f_i, which 'span' such a boundary shall be constructed by requiring that f_i's (i) evolve according to rules equivalent to those operating on bulk nodes; (ii)

are consistent with a known local velocity; (iii) conserve mass.

5.2 Aspects of Lattice Boltzmann Theory

Recalling the background theory of lattice Boltzmann set out in Chapter 4 it is appropriate to emphasise a result derived there. In what follows we stress that, the second moments of the $f_i^{(n)}$ are not zero as we have seen:

$$\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = -2c_s^2 \rho \tau S_{\alpha\beta}\,, \tag{101}$$

[31]. $S_{\alpha\beta}$ is the rate of strain tensor here. We note also that, in D2Q9, consideration of the constraints placed on the moments of the equilibrium distribution function expressed in section 4.1.2, (equations (50) and (51)) and (101) provide six equations for the nine $f_i^{(1)}$'s. Note, by combining equations (42) and (45):

$$f_i^{\dagger}(\mathbf{r},t) = f_i^{(0)}(\mathbf{r},t) + \left[f_i^{(1)\dagger} \equiv \left(1 - \frac{1}{\tau}\right) \delta_t f_i^{(1)} \right] + o\left(\delta_t^2 f_i^{(2)}\right). \tag{102}$$

5.3 Mass Conserving Lattice Closure Algorithm

The method I inherited, described in [46], allows a boundary pressure freely to develop; so boundary node densities must fluctuate. In contrast, the present method described here conserves the density associated with what are designated *fluid* links (see figure 6). This changed assumption raises key limitations in the original analysis of [46] and crucially lends the modified method to a wider range of geometries and situations (in particular, to wetting). In this section we shall treat planar and corner boundaries in separate subsections. Throughout this section denote the known boundary velocity by $\mathbf{u_0}$.

5.3.1 Planar Boundary Algorithm

Figure 6 shows an upper boundary site on the D2Q9 lattice. The dotted line shows the supposed extent of the lattice fluid. Links indexed $i = 1, 2, 3$ are cut by the line of the boundary. All non–boundary nodes are evolved according to (42). At the end of a propagation step, a *pre–collision* value of lattice momentum density, f_i, exists for all links $i \neq 5, 6, 7$ (a situation represented by use of empty circles in figure 6).

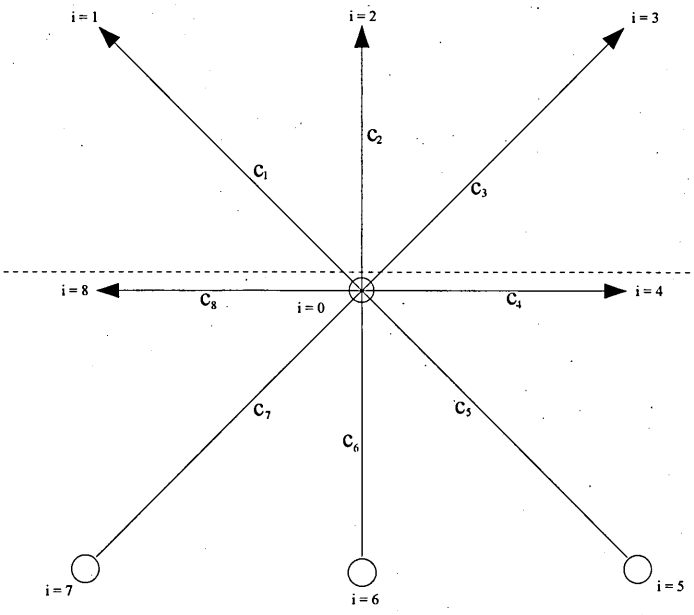

Figure 6: Schematic of a boundary node on the top, horizontal wall, immediately prior to the collision step. The wall or boundary location is denoted by the dashed line, which makes the node 'infinitesimally wet'. Propagated data is denoted by arrowheads, unknown link information (lack of data) is denoted by the open circles on links. The links are numbered as referred to in the text and link velocity vectors are numbered accordingly.

The boundary velocity provides two conditions on (moments of) the f_i's, through equations (40), but there are three post-collision $f_i^\dagger$'s which need to be obtained, namely $f_5^\dagger, f_6^\dagger$ and $f_7^\dagger$.

At the boundary node, links designated henceforth as *live* require values of f_i (links $i \neq 1, 2, 3$ in figure 6); these links connect to the flow domain and influence the evolution of adjacent nodes. The f_i's on *ghost* links propagate into the boundary.

For the geometry of figure 6, pre–collision f_i's exist for links $i = 0, 1, 2, 3, 4, 8$, post–collision $f_i^\dagger$'s are required for live links $i = 0, 4, 5, 6, 7, 8$. Local mass conservation is defined by pre(post)–collision sub–totals of the f_i's ($f_i^\dagger$'s):

$$\left(M \equiv \sum_{i \neq 5,6,7} f_i \right) = \sum_{i \neq 1,2,3} f_i^\dagger . \tag{103}$$

We construct a post–collision f_i, instantaneously correct to the accuracy of the LB model ($o(f_i^{(1)})$, for links $i = 0..8$, in the following principal steps:

1. Determine a mass conserving boundary node density, ρ' and hence the $f_i^{(0)}(\rho', \mathbf{u_0})$, $i = 0...8$. ρ' is calculated such that:

$$\sum_{i \neq 5,6,7} f_i^{(0)}(\rho', \mathbf{u_0}) = M \, , \tag{104}$$

   where M is defined in equation (103).

2. Determine pre-collision $f_i^{(1)}$'s which satisfy an under–specified system *after* equations (50), (51) and (101).

3. Collide and propagate the boundary sites according to equation (42)

Before we expand steps 1..3, it is appropriate to briefly discuss the under–specified system used for the $f_i^{(1)}$'s. Consider the $y = constant$ boundary in figure 6 (depicted by the dashed line). To conserve the total mass on only live links, it is necessary to modify the constraints on the $f_i^{(1)}$'s:

$$\sum_{i \neq 1,2,3} f_i^{(1)} = 0, \tag{105}$$

$$\sum_i f_i^{(1)} c_{i\alpha} = 0, \tag{106}$$

$$\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = -\frac{2}{3} \rho' \tau S_{\alpha\beta} \, . \tag{107}$$

The corresponding post–collision values of $f_i^{(1)\dagger}$, are all proportional (see equation (102)) to the solution of the above system. Thus, the condition of local boundary mass conservation can be stated by equation (105). In treating a forced fluid at a boundary with this method particular care must be exercised around equation (105); see sub–section 5.4 and Chapter 6.

Step 1. Evaluate M from the definition in equation (103). Since boundary velocity $\mathbf{u_0}$ is known, equations (39) and (104) provide an identity for effective density parameter ρ':

$$\rho' = \frac{M}{\sum_{i\neq 1,2,3} t_p \left[1 + \frac{\mathbf{u_0}\cdot\mathbf{c}_i}{c_s^2} - \frac{u_0^2}{2c_s^2} + \frac{(\mathbf{u_0}\cdot\mathbf{c}_i)^2}{2c_s^4}\right]} = \frac{12M}{10 + 6u_{0y} - 6u_{0y}^2}. \qquad (108)$$

It is now possible to assign $f_i^{(0)}(\rho', \mathbf{u_0})$. Note that over all links $i = 0..8$ of the boundary node, mass is not conserved. However, on links $i \neq 1, 2, 3$ there is a total mass of M which is conserved.

Step 2. Measure the local strain rates, using $o(3)$ spatially accurate finite differences of velocity. Whilst the assumed boundary velocity, $\mathbf{u_0}$ and ρ' uniquely determine $f_i^{(0)}$, $S_{\alpha\beta}$ is used separately to fix the $f_i^{(1)}$'s. The nine $f_i^{(1)}$'s, $i = 0...8$ satisfy the six equations (105)..(107) reproduced and formatted below, for convenience:

$$\begin{aligned}
f_0^{(1)} + 0f_1^{(1)} + 0f_2^{(1)} + 0f_3^{(1)} + f_4^{(1)} + f_5^{(1)} + f_6^{(1)} + f_7^{(1)} + f_8^{(1)} &= 0\,, \\
-f_1^{(1)} \qquad + f_3^{(1)} + f_4^{(1)} + f_5^{(1)} \qquad - f_7^{(1)} - f_8^{(1)} &= 0\,, \\
f_1^{(1)} + f_2^{(1)} + f_3^{(1)} \qquad - f_5^{(1)} - f_6^{(1)} - f_7^{(1)} \qquad &= 0\,, \\
f_1^{(1)} \qquad + f_3^{(1)} + f_4^{(1)} + f_5^{(1)} \qquad + f_7^{(1)} + f_8^{(1)} &= -2\rho\tau/3 \;\; S_{xx}\,, \\
f_1^{(1)} + f_2^{(1)} + f_3^{(1)} \qquad + f_5^{(1)} + f_6^{(1)} + f_7^{(1)} \qquad &= -2\rho\tau/3 \;\; S_{yy}\,, \\
-f_1^{(1)} \qquad + f_3^{(1)} \qquad - f_5^{(1)} \qquad + f_7^{(1)} \qquad &= -2\rho\tau/3 \;\; S_{xy}\,.
\end{aligned} \qquad (109)$$

The terms in the first equation with coefficient 0 are removed from the equivalent system of equations in reference [46] by our altered assumptions; the strain rates in the last three equations were measured using spatially $o(3)$ accurate finite-differences. To solve this under–determined system, select three $f_i^{(1)}$'s as free variables, in terms of whose values the remaining six basic variables are found. Certain combinations of the free $f_i^{(1)}$'s are forbidden; this may understood by regrouping quantities as follows:

$$\begin{aligned} g_1 &\equiv f_1^{(1)} + f_2^{(1)} + f_3^{(1)}, \\ g_2 &\equiv f_5^{(1)} + f_6^{(1)} + f_7^{(1)}, \\ g_3 &\equiv f_3^{(1)} + f_4^{(1)} + f_5^{(1)}, \\ g_4 &\equiv f_1^{(1)} + f_7^{(1)} + f_8^{(1)}, \\ g_5 &\equiv f_0^{(1)} + f_4^{(1)} + f_8^{(1)}. \end{aligned} \tag{110}$$

in terms of which quantities the first five of the six equations (109) become:

$$\begin{aligned} g_2 + g_5 &= 0, \\ g_3 - g_4 &= 0, \\ g_1 - g_2 &= 0, \\ g_3 + g_4 &= -2\rho\tau/3 \; S_{xx}, \\ g_1 + g_2 &= -2\rho\tau/3 \; S_{yy}. \end{aligned} \tag{111}$$

Apparently, system (111) can be solved for the $g_1..g_5$ in terms of the $S_{\alpha\beta}$. A value for g_1 (say) constrains the three $f_i^{(1)}$'s used in its definition. Forbidden sets of free $f_i^{(1)}$'s arise in this way; their elements are defined in table 2 alongside a diagrammatic interpretation. The sixth equation in the system (109) cannot be expressed in terms of $g_1..g_5$ (scrutiny of the second column in table 2 demonstrates this) and so does

not yield information.

It is very important to note that, in addition to the structure of system (111), the set of $f_i^{(1)}$'s selected as free variables depends upon which directions $\mathbf{c_i}$ contain pre–collision information.

Forbidden set of three free $f_i^{(1)}$'s	Link representation
$f_1^{(1)}, f_2^{(1)}, f_3^{(1)}$	
$f_3^{(1)}, f_4^{(1)}, f_5^{(1)}$	
$f_5^{(1)}, f_6^{(1)}, f_7^{(1)}$	
$f_1^{(1)}, f_7^{(1)}, f_8^{(1)}$	
$f_0^{(1)}, f_4^{(1)}, f_8^{(1)}$	

Table 2: Diagrammatic representation of the forbidden $f_i^{(1)}$ combinations for the planar boundary geometry of figure 6. The small (large) dot represents inclusion (exclusion) of the rest link's distribution, $f_o^{(1)}$.

In order to solve the system of equations (105)..(107), we choose free $f_i^{(1)}$'s which (i) are not a forbidden set *and* (ii) have accessible values. Criterion (ii) requires that one must select $f^{(1)}$'s corresponding to defined pre–collision f_i's; an additional reason against that third row in table 2. Here pre–collision values of $f_i'^{(1)} \approx f_i - f_i^{(0)}$, $i = 0, 1, 2$ were chosen as "free variables".

The solution for the particular boundary case represented in figure 6 now is:

$$\begin{aligned}
f_3^{(1)} &= -\kappa S_{yy} - f_1'^{(1)} - f_2'^{(1)}, \\
f_4^{(1)} &= -\kappa S_{xy} + \frac{3}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} + 2f_1'^{(1)} + f_2'^{(1)}, \\
f_5^{(1)} &= -\kappa S_{xx} + \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} - f_1'^{(1)}, \\
f_6^{(1)} &= 2\kappa S_{xx} - \kappa S_{yy} - f_0'^{(1)} - f_2'^{(1)}, \\
f_7^{(1)} &= -\kappa S_{xx} - \kappa S_{xy} + \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} + f_1'^{(1)} + f_2'^{(1)}, \\
f_8^{(1)} &= \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} - 2f_1'^{(1)} - f_2'^{(1)},
\end{aligned} \tag{112}$$

where $\kappa \equiv c_s^2 \rho \tau$.

Step 3. The pre-collision boundary node $f_i^{(1)}$'s constructed in step 2 are collided, using equation (102), then added to the corresponding $f_i^{(0)}(\rho', \mathbf{u_0})$:

$$f_i^{\dagger}(\mathbf{r}, t) = f_i^{(0)}(\rho', \mathbf{u_0}) + \left(1 - \frac{1}{\tau}\right) f_i^{(1)}. \tag{113}$$

Propagation completes the evolution of our boundary site, which apparently involves the collision of second order accurate f_i's, consistent with the target boundary velocity and the implicit wall stresses.

5.3.2 Internal Corner Algorithm

Figure 7 shows a top left internal corner boundary node. The dotted line shows the supposed location of the boundary. Quantities f_3, f_7 (links with open arrows) are ghost quantities which never participate. $f_0^{\dagger}$, $f_4^{\dagger}$, $f_5^{\dagger}$and $f_6^{\dagger}$ are required. For the geometry of figure 7 the appropriate value of M is:

$$M \equiv \sum_{i \neq 3,4,5,6,7} f_i, \tag{114}$$

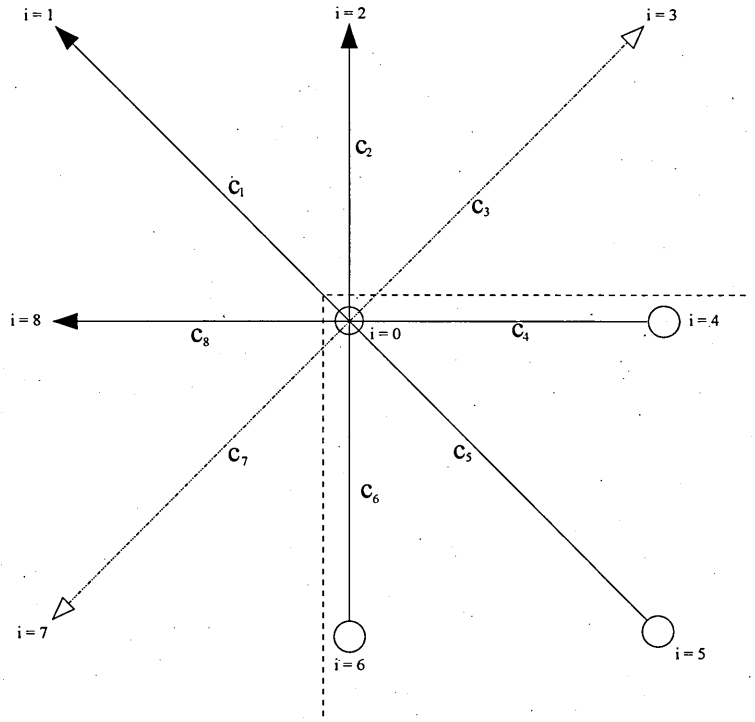

Figure 7: Schematic of a boundary node on an internal corner. The wall location is denoted by the dashed line, which makes the node 'infinitesimally wet'. Propagated data is denoted by arrowheads, unknown link information (lack of data) is denoted by the open circles on links. Ghost Links which never participate are indicated by open triangles.

and the value of ρ' is therefore:

$$\begin{aligned} \rho' &= \frac{M}{\sum_{i\neq 1,2,3,7,8} t_p \left[1 + \frac{\mathbf{u_0}\cdot\mathbf{c}_i}{c_s^2} - \frac{u_0^2}{2c_s^2} + \frac{(\mathbf{u_0}\cdot\mathbf{c}_i)^2}{2c_s^4}\right]}, \\ &= \frac{36M}{25 - 15u_{0x} + 15u_{0y} - 9u_{0x}u_{0y} - 15u_{0x}^2 - 15u_{0y}^2}. \end{aligned} \tag{115}$$

The last expression for ρ' has complicated dependence upon $\mathbf{u_0}$. However, this does not appear to affect the stability of the resulting scheme.

The mass conservation condition corresponding to equation (105) for this geometry is:

$$\sum_{i\neq 1,2,3,7,8} f_i^{(1)} = 0. \tag{116}$$

Solving for pre–collision $f_i^{(1)}$'s again follows the procedure in section 5.3.1 taking free variables to be $f_0^{(1)}$, $f_1^{(1)}$ and $f_2^{(1)}$. We arrive straightforwardly at the following solution:

$$\begin{aligned}
f_3^{(1)} &= -\kappa S_{xy} - f_1'^{(1)} - f_2'^{(1)}, \\
f_4^{(1)} &= -\frac{1}{2}\kappa S_{xx} - \kappa S_{xy} + \frac{3}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} + \frac{3}{2}f_1'^{(1)} + f_2'^{(1)}, \\
f_5^{(1)} &= -\frac{1}{2}\kappa S_{xx} + \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} - \frac{1}{2}f_1'^{(1)}, \\
f_6^{(1)} &= \kappa S_{xx} - \kappa S_{yy} - f_0'^{(1)} - f_1'^{(1)} - f_2'^{(1)}, \\
f_7^{(1)} &= -\frac{1}{2}\kappa S_{xx} - \kappa S_{xy} + \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} + \frac{3}{2}f_1'^{(1)} + f_2'^{(1)}, \\
f_8^{(1)} &= -\frac{1}{2}\kappa S_{xx} + \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} - \frac{5}{2}f_1'^{(1)} - f_2'^{(1)}.
\end{aligned} \tag{117}$$

Where, we stress that, all quantities in the right hand side are measured.

The procedure from Step 3 in section 5.3.1 may now be executed for the "free variables" of the above system. Summing the newly acquired $f_i^{(1)}$ and the $f_i^{(0)}$ components and subsequent propagation completes the evolution of the boundary site.

Note that no sets of three *obtainable* $f_i^{(1)}$'s which represent forbidden free variables arise for the internal corner geometry.

5.3.3 External Corner Algorithm

Figure 8 shows a top left external corner. The dotted line shows the supposed location of the boundary.

$f_0^\dagger$, $f_1^\dagger$, $f_2^\dagger$, $f_3^\dagger$, $f_4^\dagger$, $f_6^\dagger$, $f_7^\dagger$ and $f_8^\dagger$ are required. For M and ρ' we now have:

$$M = \sum_{i \neq 1} f_i, \tag{118}$$

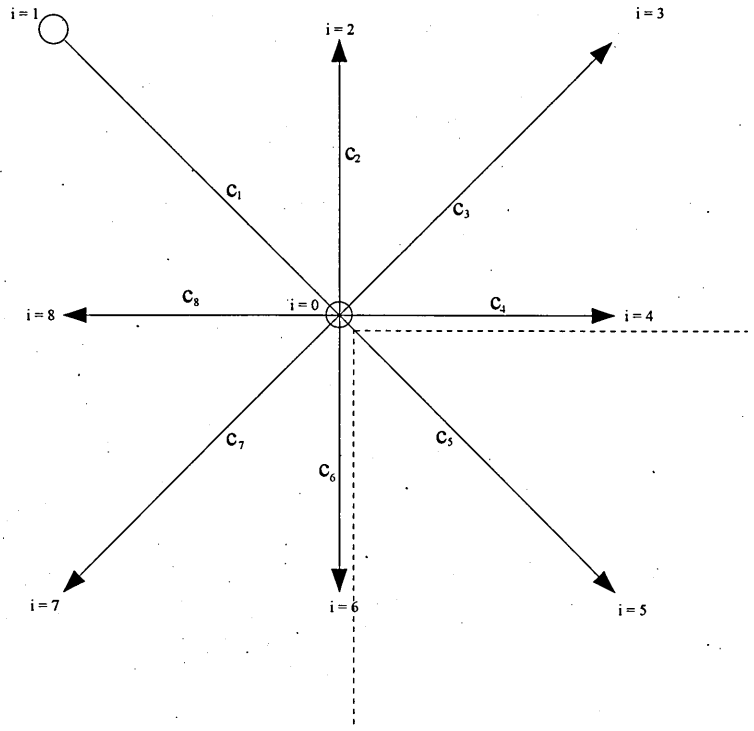

Figure 8: Schematic of a boundary node on an external corner. The wall location is denoted by the dashed lines. Propagated data is denoted by arrowheads, unknown link information (lack of data) is denoted by the open circles on links.

and;

$$\begin{aligned}\rho' &= \frac{M}{\sum_{i\neq 5} t_p \left[1 + \frac{\mathbf{u_0}\cdot\mathbf{c}_i}{c_s^2} - \frac{u_0^2}{2c_s^2} + \frac{(\mathbf{u_0}\cdot\mathbf{c}_i)^2}{2c_s^4}\right]}, \\ &= \frac{36M}{35 + 3u_{0x} - 3u_{0y} - 3u_{0x}^2 + 9u_{0x}u_{0y} - 3u_{0y}^2}. \end{aligned} \tag{119}$$

The mass conservation condition corresponding to equation (105) for this geometry is:

$$\sum_{i\neq 5} f_i^{(1)} = 0. \tag{120}$$

Solving for pre–collision $f_i^{(1)}$'s again follows the procedure in section 5.3.1 taking

free variables to be $f_0^{(1)}$, $f_3^{(1)}$ and $f_4^{(1)}$. We obtain the following solution:

$$\begin{aligned}
f_1^{(1)} &= \frac{1}{2}\kappa S_{xx} + \kappa S_{xy} - \kappa S_{yy} + \frac{1}{2}f_0'^{(1)} + \frac{3}{2}f_3'^{(1)} + \frac{3}{2}f_4'^{(1)}, \\
f_2^{(1)} &= -\frac{1}{2}\kappa S_{xx} - \kappa S_{xy} - \frac{1}{2}f_0'^{(1)} - \frac{5}{2}f_3'^{(1)} - 2f_4'^{(1)}, \\
f_5^{(1)} &= -\kappa S_{xx} - f_3'^{(1)} - f_4'^{(1)}, \\
f_6^{(1)} &= \frac{3}{2}\kappa S_{xx} + \kappa S_{xy} - \frac{1}{2}f_0'^{(1)} + \frac{3}{2}f_3'^{(1)} + \frac{1}{2}f_4'^{(1)}, \\
f_7^{(1)} &= -\frac{1}{2}\kappa S_{xx} - \kappa S_{xy} - \kappa S_{yy} + \frac{1}{2}f_0'^{(1)} - \frac{1}{2}f_3'^{(1)} + \frac{1}{2}f_4'^{(1)}, \\
f_8^{(1)} &= -\kappa S_{xx} + 2\kappa S_{yy} - f_0'^{(1)} - f_3'^{(1)} + 2f_4'^{(1)}.
\end{aligned} \tag{121}$$

The procedure from Step 3 in section 5.3.1 may now be executed for the "free variables" of the above system. Summing the newly acquired $f_i^{(1)}$ and the $f_i^{(0)}$ components and subsequent propagation completes the evolution of the boundary site.

Combinations of $f_i^{(1)}$'s now forbidden are listed in table 3. Note, other sets of three $f_i^{(1)}$'s may exist for the external corner geometry: these combinations are not listed as they rely upon three $f_i^{(1)}$'s which do not have pre–collision values.

Forbidden set of three free $f_i^{(1)}$'s	Link representation
$f_0^{(1)}$, $f_6^{(1)}$, $f_7^{(1)}$	
$f_0^{(1)}$, $f_4^{(1)}$, $f_3^{(1)}$	

Table 3: Diagrammatic representation of the forbidden combinations of $f_i^{(1)}$'s as free variables, for the external corner geometry of figure 8. Note that a small (large) dot indicates exclusion (inclusion) of the rest link. Note also that, unlike table 2, only combinations which relate to $f_i^{(1)}$'s which may be evaluated appear.

5.4 Initial Comments on Boundary Forcing

Clearly other geometries and other situations are amenable to the boundary analysis of this section. Situations in 3D, corresponding to any number of cut links are a straightforward generalization of the current method. Further extension to off–lattice boundaries is more complicated but it requires no fundamental modification to the method. The situation encountered when considering multiple immiscible fluids does, however, merit further comment.

All immiscible lattice Boltzmann fluids have interfaces supported by segregating forces impressed between different fluids. The forcing applied generally conserves mass over all lattice links $\mathbf{c_i}$, and is represented by an appropriate choice of source term, t_pG in equation (37). For lattice fluid interfaces at a boundary (the wetting problem), it is necessary to apply a segregating force to fluids at boundary fluid node. This force will not, in general, conserve mass on only the uncut links. Accordingly, it is necessary to re–state our mass conservation condition (equation (105) for the planar boundary) to apply *post–collision.* We have obtained a closure scheme which conserves mass at the boundary where the velocity is supposed to be known, but currently, no body force acts. The last assumption is, in fact, very restrictive in the context of the broad aims of this thesis. We will consider it in detail in the next chapter. For the moment, we proceed to present results obtained with the methods of this chapter.

5.5 Results

To make comparison with the popular and robust methods of on–link and mid–link bounce–back boundary conditions, steady–state duct flow was simulated. This requires $\mathbf{u_0} = \mathbf{0}$. Planar boundaries, taken to be parallel to the x–direction, were represented by (a) on–link bounce–back boundary conditions (b) mid–link bounce back (c) the inherited non mass-conserving method of [46] and (d)the method of

section 5.3.1; periodic boundary conditions were used in in the y–direction. Flow rate and hence Reynolds' number, Re, were fixed by the parameter G in equation (37). Note that $G = 0$ on the boundaries. The corresponding analytical solution is:

$$\left(v_0(y) \equiv \frac{G}{6\rho\nu} y(W-y), 0\right). \tag{122}$$

The scalar difference between the measured steady-state flow profile, $(v_m(y), 0)$:

$$\Delta_W \equiv \sum_{y \neq 0} \left| \frac{v_m(y) - v_0(y)}{v_0(y)} \right|, \tag{123}$$

was measured at constant $Re = 0.5$ for a range of lattice resolutions. The latter was parameterised by the duct width $W = 10 \times 2^N$, $N \leq 5$.

In general, a plot of measured error (Such as that defined in equation (123)) as a function of resolution has a (negative) gradient. The magnitude of which defines the order of the accuracy of the scheme: The steeper the graph of Δ_W versus resolution, the more accurate the scheme.

The data of figure 9 shows the value of Δ_W as a function of resolution for boundary methods (a)..(d). The short–comings of the first-order accurate on–link bounce–back technique ((a)) in terms of the reduced accuracy of the solution close to the lattice boundaries and first order–accuracy are documented elsewhere [46]. Mid–link bounce–back ((b)), the inherited method of [46] and method of section 5.3.1 ((c)) are all seen to be second order accurate in spatial gradient (from the gradient in figure 9), with the latter two methods ((c) and (d)) slightly more accurate and, of course, capable of representing a boundary with any velocity $|\mathbf{u_0}| < 0.1$ and time dependence.

To demonstrate that our boundary closure is robustly correct, even at short times, and that it accommodates moving boundaries we considered a lattice fluid initially at rest ($f_i(\mathbf{r}, 0) = f_i^{(0)}(\rho, \mathbf{0})$ everywhere) between two horizontal planar

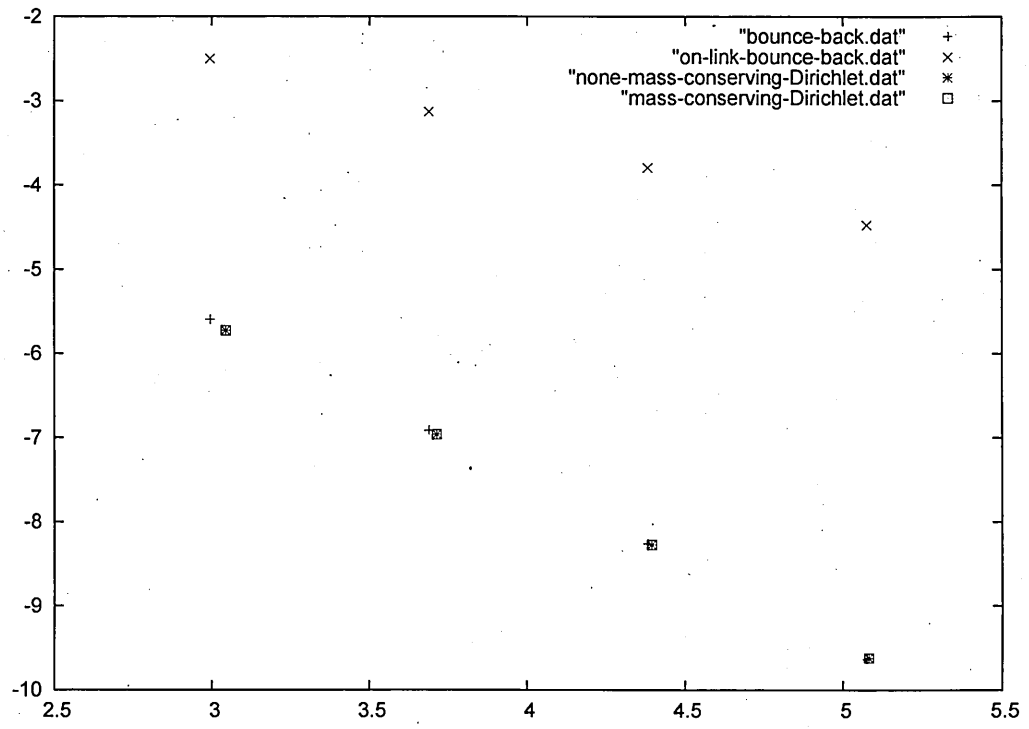

Figure 9: Error, ΔW, defined in equation (123), as a function of lattice resolution. The latter is measured by lattice width W. $Re = 0.5$. The diagonal crosses represent bounce–back boundary conditions, the erect crosses represent mid–link bounce back, the star symbols represent the inherited non-mass conserving version of the bounceback condition and the open squares represent the lattice closure strategy developed in section 5.3.1.

boundaries $y = 0^-, W^+$; the latter was impulsively started from rest at $t = 0^+$ to velocity $(u_0, 0)$. The analytic solution to this transient flow is:

$$v_x(y,t) = \frac{u_0}{W}\, y + \frac{2u_0}{\pi} \sum_m \frac{(-1)^m}{m} \sin\left(\frac{m\pi y}{W}\right) \times e^{-m^2\pi^2\bar{t}}\ , \qquad \bar{t} = \frac{\nu t}{W^2}\,. \tag{124}$$

Where note "m" is a dummy index. Here $\bar{t}$ represents a *dimensionless* time parameter. Snapshots of the developing flow across the whole width of the duct, obtained for discrete times $t = 10$, 100, 1000 with corresponding collision parameter, $1/\tau = 0.6$, 1.0, 1.6, were collected. Results with the present method are identical to those obtained using the original method [46]; remarks made in [46] still apply.

To assess the accuracy of the corner boundaries a square, 2D lid–driven cavity was simulated over a small range of $Re \approx 93$ (where a secondary vortex is known first to appear) and resolutions. See figure 10 caption for simulation data. Figure 10 shows the stream function obtained with our new boundary conditions enforced over the entire boundary; in particular the top corners of the enclosure were assumed to be moving with the lid speed. Figure 11 shows detail of the stream function in the bottom left corner of the flow. Figure 12 shows the y–component of velocity along the right hand boundary; the point of detachment is clearly visible in this data.

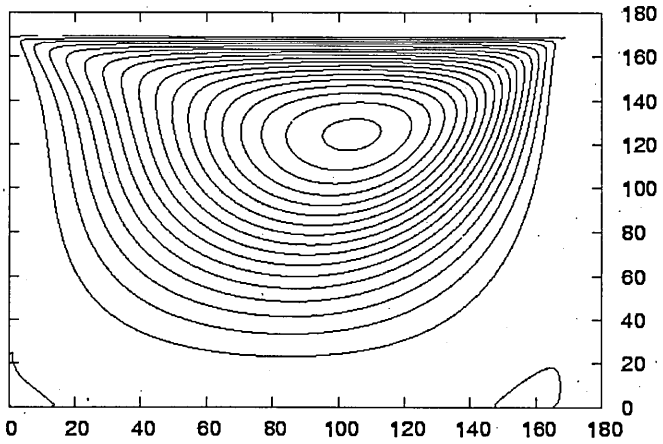

Figure 10: Steady–state stream function for a 2D square lid–driven cavity. Data corresponds to a square box 256×256. The single relaxation parameter was $\tau = 3.33$; the density parameter $\rho = 1.8$. For the data show $Re = 100$. The four internal corners and the planar boundaries were simulated using the method of section 5.3.1; corners in contact with the moving (top) lid were moved with lid velocity.

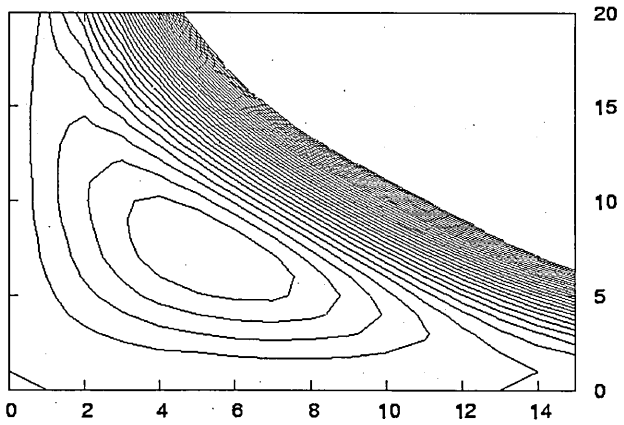

Figure 11: Detail of the steady–state stream function for a 2D square lid–driven cavity, shown in figure 10. This image corresponds to the detached flow due to the secondary vortex encountered in the bottom left corner.

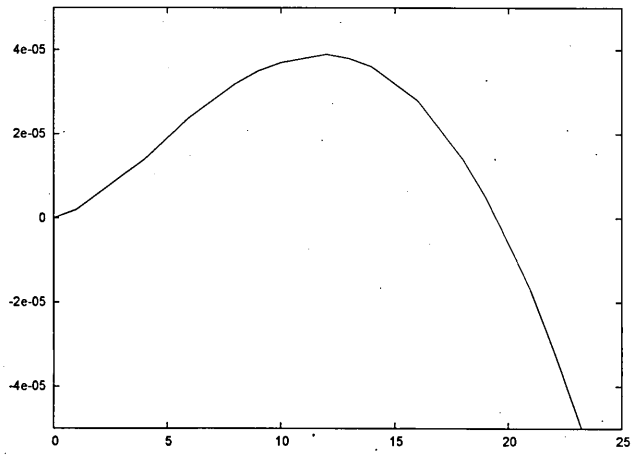

Figure 12: The y–velocity distribution along the right–hand boundary of the simulation box; the change in the sign of the velocity corresponds to the point in which the stream function detaches

5.6 Comments

In this chapter we have set-out and validated a very adaptable method for closing a lattice Boltzmann simulation lattice by calculating appropriate values for the set of missing momentum densities, subject to the condition of local mass conservation. Like the previous algorithm, which allowed pressure adaptation at the boundary [46], the method presented here is demonstrably accurate, flexible and instantaneously correct. However, the present method can also apply (i) to an increased range of geometries other than the set presented here and (ii) to boundaries where fluid forces need to be impressed- in particular the case of immiscible fluid–fluid interfaces at a boundary as carried out in Chapter 6.

It is noted that the method of this chapter may be generalised for application to 3D situations.

Calculations have been shown here for application of the method to external corners. Due to time constraints both internal and external corners were not simulated, however the case of the internal corner was deemed to be most algorithmically demanding and was therefore the situation simulated. Brief investigation into the application of the external corner algorithm appear satisfactory.

Furthermore, we find that the scheme devised in this chapter remains stable over a range of collision parameter. Complex simulations with the resolution and the Re of that used to generate the data of figures 10...12 show results which do not change appreciably for $2.5 > \tau > 0.56$. Time and processing constraints prevent a full consideration of the stability of our closure scheme with variation of τ but it is appropriate to emphasize that e.g. bounce back method accuracy varies considerably with τ as τ is shifted away from its optimum value of 1.0.

The key restriction on the analysis of this chapter is that the fluid at the boundary was assumed to be free of any force. In the next chapter I shall (i) revise this assumption and (ii) apply the method to produce a model of fluid slip.

6 Mass Conserving Boundary Closure Scheme for Fluids Subject to External Forces

The algorithm after Chapter 5 has been demonstrated to treat the lattice fluid on boundary and bulk nodes to the same accuracy and in a demonstrably equivalent manner. Whilst the new method presented here applies to rectangular geometries, it adds to our previous method of Chapter 5 the ability to subject boundary fluid to an external force. Though this is much more intricate this will become very important in Chapter 10.

For the purposes of clarity and completeness some of the analysis of the previous chapters is reiterated here to produce a coherent chapter in its own right. For the reader who is not solely reading this chapter, and has fully engaged with the material previously, it may be appropriate to omit certain sections. However the intricate nature of the analysis to be presented here makes me reluctant to recommend any omissions.

6.1 Introduction

This chapter is an extension of previous work outlined in chapter 5 which in turn is an extension to the inherited algorithm of [46], the aim is to facilitate (i) Navier-type boundary conditions and (ii) simulate boundary fluids under external forces, such as are applied to produce fluid-fluid interfaces; the essentials of multi–component LB boundary conditions for e.g. wetting (this will be addressed in Chapter 10). Like Chapter 5, the work presented in this section is designed to give boundary fluid in transient flow near the boundary the same dynamics and accuracy of representation as the bulk lattice fluid (at all times), it applies transparently to complex boundary geometries (corners) with no diminution of accuracy and it admits of the representation of moving fluid boundaries, including those crossed by fluid.

If a fluid is said to stick; to have 'no-slip' against solid surfaces; then $\mathbf{u}_0(\mathbf{r}', t)$ corresponds to the velocity of any solid boundaries. However, there have been persistent doubts about the no-slip condition for more than a century. Take a topical example relevant to multi–component, microfluidic flow of Newtonian fluids; a fluid-fluid interface in motion relative to a solid boundary. The fluid close to a so–called dynamic contact line (DCL) must clearly violate the no–slip condition. For the case of Couette flow, (viscous laminar flow in a channel where the flow is driven by the motion of one of the boundaries relative to the other), one simple theory is the Navier slip theory briefly outlined in Chapter 4, which relates the boundary–tangent component of fluid velocity, the slip velocity, v_s, to the tangential component of fluid shear at the surface:

$$v_s(\mathbf{r}', t) = b \left[\frac{\partial v_t}{\partial x_n} \right]_{\mathbf{r}' t}, \tag{125}$$

in which the constant b is the *slip length* which is for present purposes, taken to be a molecular property of the material. The boundary–normal component of fluid velocity is equal to the boundary–normal component of the boundary velocity.

It is stressed at the outset that the purpose of this section is not to investigate the origin or applicability of particular boundary conditions; rather to derive and evaluate a single tractable, practical, simple–to–implement and versatile LB closure step. Accordingly, the method assumes a known boundary velocity and after that in equation (125), a local mass conservation (essential to the treatment of multi–component fluids at a boundary) and a known external force distribution at the boundary. After reviewing salient theory in section 6.2, we consider a flat boundary at which fluid is moving with specified velocity, subject to a specified, external body force, in section 6.4.1. We summarize equivalent results for internal and external corners in sections 6.4.2 and 6.4.3. To evaluate our method, section 6.5 presents accuracy plots, profiles of solved, pressure driven duct flows with moving and stationary boundaries and stream functions.

6.2 Further Background to LB Dynamics

It is appropriate to recall that collision and propagation of LB's primary quantity, the link–based momentum distribution function or density, $f_i(\mathbf{r},t)$, $i = 0..8$ is expressed in a kinetic equation:

$$f_i(\mathbf{r}+\mathbf{c}_i\delta_t, t+\delta_t) \equiv f_i^{\dagger}(\mathbf{r},t) = f_i(\mathbf{r},t) + \omega\left(f_i^{(0)}(\rho,\mathbf{v}) - f_i(\mathbf{r},t)\right) + \phi_i, \qquad (126)$$

where δ_t represents the time step and $f_i^{\dagger}(\mathbf{r},t)$ is the post–collision, pre–propagate value of the momentum density. The vectors and indexing of the D2Q9 velocity basis $\mathbf{c}_i$ are defined in figure 13. The so called source term, ϕ_i, in equation (126) will be discussed shortly. Essentially it is used to impress a body-force density on the fluid.

Following the constraints underlying the distribution function, shown in equations (50) and (51) we may write the following condition:

$$\sum_i f_i^{(n>0)}\Delta_i = 0, \qquad (127)$$

where $\Delta_i = 1, c_{ix}, c_{iy}$.

Recall that equation (60), from the Champman-Enskog expansion shown in section 4.1.2, had the second moments of the $f_i^{(1)}$ not equal to zero;

$$\partial_{t1}\Sigma_i f_i^{(0)} c_{i\beta} + (1-\frac{1}{2\tau})\Sigma_i(\partial_{t0} + c_{i\alpha}\partial_\alpha) f_i^{(1)} c_{i\beta} = -\frac{1}{\tau}\Sigma_i f_i^{(2)} c_{i\beta}. \qquad (128)$$

In fact provide a local expression for the lattice fluid rate of strain, see equation (84):

$$\frac{1}{2}\sum_{ij}\left(\frac{1}{\tau} - 2\right) f_i^{(1)} c_{i\alpha} c_{i\beta} = 2\nu\rho S_{\alpha\beta}\,, \qquad (129)$$

where recall $S_{\alpha\beta} = 1/2(\partial_\alpha v_\beta + \partial_\beta v_\alpha)$. It is convenient to re–express equation (129):

$$-(1 - \frac{1}{2\tau}) \sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = 2\nu\rho S_{\alpha\beta}. \tag{130}$$

All the equations written so far in this chapter describe a fluid in two or three dimensions. In D dimensions, equations (130) number $D(D+1)/2$ and equations (127) number $(D+1)$; in all systems (130) and (127) provide $(D+1)(D/2+1)$ equations for Q $f_i^{(1)}$'s.

Recall from Chapter 5 that in the case of a D2Q9 LBGK model, equations (127) and (130) yield a total of 6 equations for $Q = 9$, $f_i^{(1)}$'s; moreover, we can use equations (35) and (36) to obtain from equation (130) the following:

$$\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = -2c_s^2 \rho\tau S_{\alpha\beta}. \tag{131}$$

Using equations (126) and (45) (with $n < 2$), it is possible to express the collision part of the evolution as a relaxation of (just) the $f_i^{(1)}$ component of $f_i(\mathbf{r}, t)$:

$$f_i^\dagger(\mathbf{r}, t) = f_i^{(0)}(\rho, \mathbf{v}) + \left[f_i^{(1)\dagger} \equiv \delta_t \sum_i \omega f_i^{(1)} \right] + \phi_i + o\left(\delta_t^2 f_i^{(2)}\right), \tag{132}$$

that is, collision may be represented by a replacement:

$$f_i^{(1)} \to f_i^{(1)\dagger}, \qquad f_i^{(1)\dagger} = \sum_i (\delta_{ij} - \Omega_{ij}) f_j^{(1)}, \tag{133}$$

and the addition of source term, ϕ_i; note that we have set $\delta_t = 1$ here for simplicity. In the current context, using equation (35), equation (133) simplifies to make LBGK an attractive option:

$$f_i^{(1)\dagger} = \left(1 - \frac{1}{\tau}\right) f_i^{(1)}. \tag{134}$$

The term ϕ_i in equations (126) and (132) is a source term used to impress an external force on the lattice fluid at the macroscopic, Navier–Stokes level (for exam-

ple to represent a uniform pressure gradient or, in the case of multiple immiscible fluids, the effects of interfacial tension [3]). For current purposes, we neglect spatial variation in the macroscopic force; however this assumption will be revisited in chapter 10, taking the macroscopic momentum equation corresponding to equation (126) to be:

$$\frac{\partial}{\partial t}\rho v_\alpha + \frac{\partial}{\partial x_\beta}\rho v_\beta v_\alpha = -\frac{\partial}{\partial x_\alpha}\rho + \frac{\partial}{\partial x_\beta}\left(2\rho\nu S_{\alpha\beta}\right) + \sum_i \phi_i c_{i\alpha}, \tag{135}$$

in which all other symbols and notations have their usual meaning. The relationship between the ϕ_i's and a non–uniform macroscopic force has been considered by Ladd and Verberg [47] and by Guo, Zheng and Shi [48]; This issue shall be further considered in chapters 7 and 8. Note that forcing a fluid must conserve mass over all nodal lattice links $i = 0..(Q-1)$:

$$\sum_{i=0}^{(Q-1)} \phi_i = 0. \tag{136}$$

Without considering MCLB in too much detail at this stage (as multiphase extensions are not rigorously covered until chapter 7), it is appropriate to note that all multi–component lattice Boltzmann fluids have interfaces produced by phase index coupled pressure tensor fluctuations within what is a single effective fluid [3]. These fluctuations must, in the end, be considered as (segregating) external forces, impressed in certain regions of a single fluid. Such interface forcing is represented by an appropriate choice of source term ϕ in equation (126) and, in bulk simulations or with periodic boundaries, mass (but not momentum) is conserved. Other situations involve external forces at the boundary; for example a fluid under gravity.

A problem arises for LB fluids under external forces at the boundary. For a lattice fluid interface at a boundary, it is essential to apply a segregating force; more generally, if bulk fluid is subject to any external force, then the ideally equivalent

boundary fluid should also be under the same external force as we consider boundary nodes to be infinitesimally wet. In the present context, this force is assumed to be known; the problem we seek to address is that the corresponding source term does not, in general, conserve mass on only uncut links; put more precisely, partial totals of the interface source term are not necessarily zero:

$$\Sigma_i' \phi_i \neq 0, \tag{137}$$

where the symbol Σ_i' indicates a summation over a restricted range of subscript i, which excludes the value of i corresponding to cut links.

6.3 Limiting Assumptions

In this chapter we will concentrate on a methodology for treating **constant** forces. The limitations of this assumption will become apparent during chapters 7 and 8, where it will be very obvious that when we deal with interfacial tension, we are concerned with variable forces.

However, with the methodology of the present chapter in place, we shall find, in Chapter 10, that a very simple re-definition of the strain rate will be sufficient to generalise all the results and methods of this chapter to multi–component variable forces at a boundary, in particular see section 10.5.

6.4 Lattice Closure Algorithm for Fluid at a Boundary Subject to a Constant External Force

Mass conservation is a widely-used and convenient assumption when closing LB simulations with Dirichlet boundary conditions; the same assumption becomes unavoidable when multiple fluids are in contact at a solid boundary. Moreover, in such situations (interfaces in contact, *wetting* a boundary) it is necessary, in LB

simulations, to impress an external force on the boundary fluid, in order to create interfacial tension. Hence there is a need to develop a lattice closure for a specified boundary velocity (which may correspond to a slip velocity) with mass conservation, which allows for the presence of a force. Other situations to which such a closure might apply are fluid under gravity or fluid with a separately calculated pressure distribution.

Throughout this analysis it shall be assumed that an effective boundary velocity is always known. This velocity may correspond to a fluid slip (pre-calculated according to some assumed model *e.g.* equation (125)).

In summary, the aim is to calculate momentum densities, f_i, for links of a boundary node which (i) evolve according to rules equivalent to those governing bulk nodes, (ii) are consistent with a known, local, boundary velocity (including a slip velocity), (iii) are consistent with a known local external force and (iv) will conserve mass locally.

6.4.1 Planar Boundary Algorithm

Whilst keeping the overall approach as general as possible, it will be necessary, for definiteness, to refer to a particular lattice and flow geometry. For this purpose and for consistency with other sections we consider a planar boundary site on the LBGK D2Q9 lattice shown in figure 13 as with Chapter 5. The dotted line indicates the uppermost extent of the lattice fluid. Links $i = 1, 2, 3$ are cut. At the end of a propagation step, a *pre-collision* value of f_i, exists for all *incoming* links $i \neq 5, 6, 7$; a situation represented by use of empty circles. In contrast to chapter 5 we now suppose it is necessary impress an external force on the boundary fluid; that is, to use a source term ϕ_i in equation (126). Say that the node of figure 13 bounds a flow driven by a uniform, external body force in the x-direction, representing gravity or a uniform pressure gradient; this force is obtained from a source term $\phi_i = t_p G c_{ix}$.

As with Chapter 5, we denote as *live* those links which require post-collision

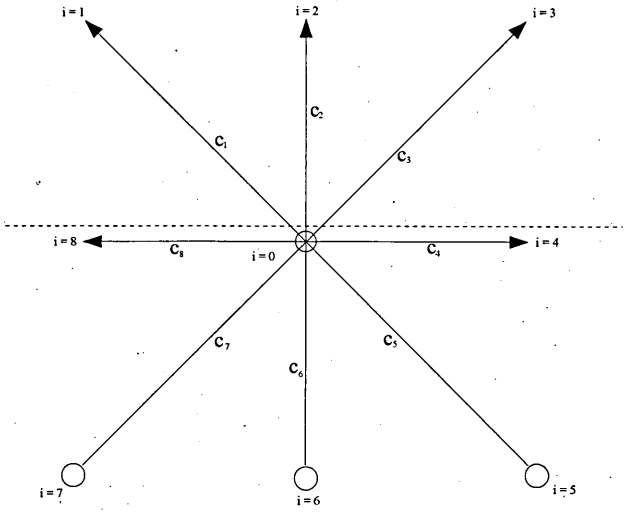

Figure 13: A boundary node on a horizontal wall, immediately prior to a collision step. Links $i = 1, 2, 3$ ($i = 0, 4..8$) all point into solid (fluid). The wall location is denoted by the dashed line. Propagated link data is denoted an arrowhead on the link, lack of data by an open circle..

values of $f_i^\dagger$; links $i \neq 1, 2, 3$ in figure 13 are live; they connect to the flow domain. The f_i's on *dead* links propagate into the boundary. Note that fig 13 is essentially identical to that of Chapter 5, figure 13 is inserted for convenience only. For the geometry of figure 13, pre–collision f_i's exist on *incoming* links $i = 0, 1, 2, 3, 4, 8$; post–collision $f_i^\dagger$'s are required only for live links $i = 0, 4, 5, 6, 7, 8$.

Boundary mass conservation is defined in terms of equality between sub–totals of the f_i's and (different) sub-totals of the $f_i^\dagger$'s. Recall that for the geometry of figure 13, local mass conservation is expressed:

$$\left(M \equiv \sum_{i \neq 5,6,7} f_i \right) = \sum_{i \neq 1,2,3} f_i^\dagger, \tag{138}$$

of which the equivalent formula in $D3QN$ is a straightforward generalisation.

Applying a force on boundary nodes does not conserve mass on the sub-set of

live links. Again for the example geometry of figure 13, an amount of mass:

$$\delta M = \sum_{i \neq 1,2,3} \phi_i, \tag{139}$$

is introduced into the flow domain by the macroscopic force.

In general, nodes are thought of as being evolved by collision and propagation, with collisions described by equation (132). Constructing post–collision, boundary f_i's, instantaneously correct to the accuracy of the bulk LB model essentially in 4 steps:

1. Determine M and δM from generalized forms of equations (138) and (139):

$$M = \sum_{incoming} f_i, \tag{140}$$

and;

$$\delta M = \sum_{live} \phi_i. \tag{141}$$

2. Determine the $f_i^{(0)}(\rho', \mathbf{u_0})$ from an *effective* density, ρ' such that;

$$\sum_{live} f_i^{(0)}(\rho', \mathbf{u_0}) = M - \delta M\,, \tag{142}$$

where the summation is taken over values of i corresponding to live links and M and δM are defined in equations (140) and (141) (or, alternatively, after equations (138) and (139)).

3. Determine pre-collision $f_i^{(1)}$'s from an under–specified system after equations (127) and (131) in an identical manor to that carried out in chapter 5.

4. Collide the boundary sites according to equation (132) after Chapter 5 but with addition of the source term ϕ_i, note.

Before detailing the above steps, a few remarks are in order. It is important to note that the condition of boundary mass conservation is satisfied only after the completion of final step (4). Also note that the under–specified system solved for the $f_i^{(1)}$'s in step (3) is an adaptation of $(D+1)(D/2+1)$ equations (127) and (131), namely:

$$\sum_{live} f_i^{(1)} = 0, \tag{143}$$

$$\sum_{i} f_i^{(1)} c_{i\alpha} = 0, \tag{144}$$

$$\sum_{i} f_i^{(1)} c_{i\alpha} c_{i\beta} = 2\rho' \nu S_{\alpha\beta}\,. \tag{145}$$

We note that in the above system of equations (143) and (144) are independent of particular LB model, however the $D(D+1)/2$ equations (145) depend upon model through the Ω_{ij}. We note that the corresponding, post-collision, values $f_i^{(1)\dagger}$ are a simple linear combination of the solution of equations (143)..(145) above (see equation (133)). Finally we emphasise that it is only after the completion of step 4 that the mass on live simulation links balances mass which initially propagated onto the boundary node.

Step 1. Evaluate M and δM using the definitions in equations (138) and (139) respectively.

Step 2 The boundary (or slip) velocity $\mathbf{u_0}$ is known. Equations (38) and (142) provide an identity for effective boundary node density, ρ':

$$\rho' = \frac{M - \delta M}{\sum_{live} P^{(0)}\left(\mathbf{u_0} \cdot \mathbf{c}_i\right)}. \tag{146}$$

For the example geometry of figure 13 and the corresponding LBGK equilibrium, equation (146) is:

$$\rho' = \frac{M - \delta M}{\sum_{i \neq 1,2,3} t_p \left[1 + \frac{\mathbf{u_0}\cdot\mathbf{c}_i}{c_s^2} - \frac{u_0^2}{2c_s^2} + \frac{(\mathbf{u_0}\cdot\mathbf{c}_i)^2}{2c_s^4}\right]} = \frac{6(M - \delta M)}{5 + 3u_{0y} - 3u_{0y}^2}, \tag{147}$$

in which, note, velocity component u_{0x} does not appear. To evaluate δM from equation (141) (or equation (139)) requires the values of live source terms, ϕ_i. For our chosen example (a uniform body force in the x-direction, recall) $\phi_i = t_p G c_{ix}$:

$$\delta M = G \sum_{i \neq 1,2,3} t_p c_{ix}. \tag{148}$$

It is now possible to assign ρ' from equation (147) and hence $f_i^{(0)}(\rho', \mathbf{u_0})$.

Step 3. The calculation of the $f_i^{(1)}$ component follow an identical procedure to that of "step 2" in section 5.3.1. For the purposes of producing a coherent and self contained account, the calculation of $f_i^{(1)}$'s is documented here. For the reader who has fully engaged with the concepts and methods of "step 2" in section 5.3.1 you may wish to bypass this step and continue to "step 4".

For our chosen example boundary, flow and model equations (143), (144) and (145) yield:

$$\sum_{i \neq 1,2,3} f_i^{(1)} = 0, \tag{149}$$

$$\sum_i f_i^{(1)} c_{i\alpha} = 0, \quad \alpha = x, y, \tag{150}$$

$$\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = -\frac{2}{3} \rho' \tau S_{\alpha\beta} \quad \alpha = x, y, \quad \beta = x, y, \tag{151}$$

in the last of which we have used equations (130) and (35). Here then, nine $f_i^{(1)}$, $i = 0...8$ satisfy the six equations (149)..(151) re–formatted below:

$$\begin{aligned}
f_0^{(1)} + 0f_1^{(1)} + 0f_2^{(1)} + 0f_3^{(1)} + f_4^{(1)} + f_5^{(1)} + f_6^{(1)} + f_7^{(1)} + f_8^{(1)} &= 0, \\
f_0^{(1)} \qquad\qquad\qquad\quad + f_4^{(1)} + f_5^{(1)} + f_6^{(1)} + f_7^{(1)} + f_8^{(1)} &= 0, \\
-f_1^{(1)} \qquad + f_3^{(1)} + f_4^{(1)} + f_5^{(1)} \qquad - f_7^{(1)} - f_8^{(1)} &= 0, \\
f_1^{(1)} + f_2^{(1)} + f_3^{(1)} \qquad - f_5^{(1)} - f_6^{(1)} - f_7^{(1)} \qquad &= 0, \\
f_1^{(1)} \qquad + f_3^{(1)} + f_4^{(1)} + f_5^{(1)} \qquad + f_7^{(1)} + f_8^{(1)} &= -2\rho\tau/3 \; S_{xx}, \\
f_1^{(1)} + f_2^{(1)} + f_3^{(1)} \qquad + f_5^{(1)} + f_6^{(1)} + f_7^{(1)} \qquad &= -2\rho\tau/3 \; S_{yy},
\end{aligned}$$

$$-f_1^{(1)} \qquad +f_3^{(1)} \qquad -f_5^{(1)} \qquad +f_7^{(1)} \qquad = -2\rho\tau/3 \; S_{xy} \; .$$

(152)

Note, the surface strain rates in the right hand side of last three equations must all be determined, in this work a spatially $o(3)$ accurate forward and backward finite-difference method is utilised. Note that in the case of slip, these finite difference expressions, which involve the instantaneous surface slip velocity, were assumed simply to assign a quasi-statically developing slip velocity at the next time step.

When applying this LB method to other models, differences must be assumed to arise a this point. For a 3D LBGK model, a system similar to equations (129) results; for general LB models the system of equations corresponding to equations (152) would contain coefficients of the $f_i^{(1)}$'s which are linear combinations of the Ω_{ij}'s (see e.g. equation (129)).

Following the parallel analysis of Chapter 5, to solve the under-determined system of equations (152), select a triplet of known $f_i^{(1)}$'s as free variables. Certain triplets of the free $f_i^{(1)}$'s are forbidden, which may understood as follows. Defining quantities:

$$\begin{aligned}
g_1 &\equiv f_1^{(1)} + f_2^{(1)} + f_3^{(1)}, \\
g_2 &\equiv f_5^{(1)} + f_6^{(1)} + f_7^{(1)}, \\
g_3 &\equiv f_3^{(1)} + f_4^{(1)} + f_5^{(1)}, \\
g_4 &\equiv f_1^{(1)} + f_7^{(1)} + f_8^{(1)},, \\
g_5 &\equiv f_0^{(1)} + f_4^{(1)} + f_8^{(1)},
\end{aligned} \tag{153}$$

the first five of the six equations (152) may be written:

$$\begin{aligned}
g_2 \qquad\qquad + g_5 &= 0, \\
g_3 - g_4 \qquad\quad &= 0, \\
g_1 - g_2 \qquad\qquad &= 0, \\
g_3 + g_4 \qquad\quad &= -2\rho\tau/3 \; S_{xx}, \\
g_1 + g_2 \qquad\qquad &= -2\rho\tau/3 \; S_{yy}.
\end{aligned} \tag{154}$$

Which system (154) may be solved for the $g_1..g_5$ in terms of the $S_{\alpha\beta}$. The value of (say) g_1 so obtained constrains the sum $f_1^{(1)} + f_2^{(1)} + f_3^{(1)}$; accordingly $f_1^{(1)}$, $f_2^{(1)}$ and $f_3^{(1)}$ cannot all be assigned; cannot all be free variables. Forbidden triplets of free $f_i^{(1)}$'s for our example system are defined in table 4, alongside a diagrammatic interpretation. Note, the sixth equation in the system (152) cannot be expressed in terms our chosen $g_1..g_5$.

Above *known* $f_i^{(1)}$'s were referred to. In addition to the structure of system (154), the set of three $f_i^{(1)}$'s selected as free variables should depend upon which links, $\mathbf{c_i}$, contain information originating from within the flow domain. To solve equations (149)..(151) then, free variable $f_i^{(1)}$'s are used which (i) do not comprise a forbidden set and (ii) have accessible values.

Free variables are chosen as in the solution equations (149)..(151):

$$f_i^{(1)} \approx \left(f_i - f_i^{(0)}(\rho', \mathbf{u_0}) \right), \quad i = 0, 1, 2, \tag{155}$$

Forbidden set of three free $f_i^{(1)}$'s	Link representation
$f_1^{(1)}, f_2^{(1)}, f_3^{(1)}$	
$f_3^{(1)}, f_4^{(1)}, f_5^{(1)}$	
$f_5^{(1)}, f_6^{(1)}, f_7^{(1)}$	
$f_1^{(1)}, f_7^{(1)}, f_8^{(1)}$	
$f_0^{(1)}, f_4^{(1)}, f_8^{(1)}$	

Table 4: Diagrammatic representation of the forbidden $f_i^{(1)}$ combinations for the planar boundary geometry of figure 13. The large (small) dot represents inclusion (exclusion) of the rest link.

in terms of which the following is a solution of system of equations (152)

$$\begin{aligned}
f_3^{(1)} &= -\kappa S_{yy} - f_1'^{(1)} - f_2'^{(1)}, \\
f_4^{(1)} &= -\kappa S_{xy} + \frac{3}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} + 2f_1'^{(1)} + f_2'^{(1)}, \\
f_5^{(1)} &= -\kappa S_{xx} + \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} - f_1'^{(1)}, \\
f_6^{(1)} &= 2\kappa S_{xx} - \kappa S_{yy} - f_0'^{(1)} - f_2'^{(1)}, \\
f_7^{(1)} &= -\kappa S_{xx} - \kappa S_{xy} + \frac{1}{2}\kappa S_{yy} + \frac{1}{2}f_0'^{(1)} + f_1'^{(1)} + f_2'^{(1)}, \\
f_8^{(1)} &= \kappa S_{xy} - \frac{1}{2}\kappa S_{yy} - \frac{1}{2}f_0'^{(1)} - 2f_1'^{(1)} - f_2'^{(1)},
\end{aligned} \tag{156}$$

where:

$$\kappa \equiv c_s^2 \rho \tau. \tag{157}$$

The choice of free variables $f_i^{(1)}$ made here, $i = 0, 1, 2$ is not unique. This shall be further discussed in section 6.5.

Step 4. The pre-collision boundary node $f_i^{(1)}$'s constructed in step 3 are collided,

using equation (132), then added to the corresponding $f_i^{(0)}(\rho', \mathbf{u_0})$, yielding a post–collision momentum density:

$$f_i^{\dagger}(\mathbf{r}, t) = f_i^{(0)}(\rho', \mathbf{u_0}) + \sum_i (\delta_{ij} - \Omega_{ij}) f_j^{(1)} + \phi_i, \tag{158}$$

which, for the example LBGK system, reduces to:

$$f_i^{\dagger}(\mathbf{r}, t) = f_i^{(0)}(\rho', \mathbf{u_0}) + \left(1 - \frac{1}{\tau}\right) f_i^{(1)}(\rho, \mathbf{u_0}) + \phi_i. \tag{159}$$

Before proceeding to consider other boundary shapes (corners), a few remarks are in order. Note that the analysis of this section is valid, stability issues notwithstanding, for any value of collision parameter τ. Of course, as with Chapter 5, the choice of the free variables in step 3 of this section should not affect the accuracy or stability of the resulting closure scheme. This shall be further discussed in section 6.5.

6.4.2 Internal Corner Algorithm

We illustrate the versatility of the method outlined in section 6.4.1 by treating, albeit in less detail, internal and external corners. Consider our example D2Q9 LBGK system.

Figure 14 shows a top left internal corner boundary node. The dotted line shows the supposed location of the boundary.

Quantities f_3, f_7 (links represented by broken lines) are *ghost* (not, note, dead) quantities which never participate. $f_0^{\dagger}$, $f_4^{\dagger}$, $f_5^{\dagger}$ and $f_6^{\dagger}$ are required. For the geometry of figure 14 the appropriate value of M is:

$$M \equiv \sum_{i \neq 3,4,5,6,7} f_i, \tag{160}$$

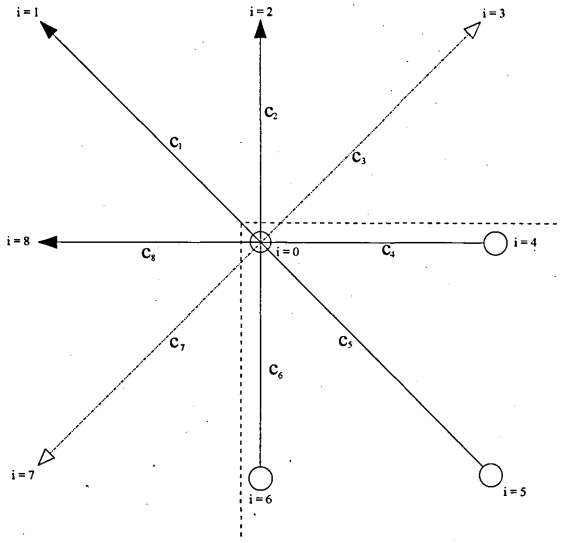

Figure 14: A boundary node on an internal corner. The walls locations are denoted by the dashed lines. The node is taken to lie in the fluid with the rest link. Propagated link data is denoted by a solid arrowhead or solid square on the link, unavailable link information (lack of data) by an open circle. Ghost links, which never participate, are indicated by open arrowheads.

and the value of ρ' is therefore given by:

$$\rho' = \frac{M - \delta M}{\sum_{i \neq 1,2,3,7,8} t_p \left[1 + \frac{\mathbf{u_0} \cdot \mathbf{c}_i}{c_s^2} - \frac{u_0^2}{2c_s^2} + \frac{(\mathbf{u_0} \cdot \mathbf{c}_i)^2}{2c_s^4}\right]}, \tag{161}$$

where δM is the mass associated with the source terms to be added on the live links (see equation (139)):

$$\delta M \equiv \sum_{i \neq 1,2,3,7,8} \phi_i. \tag{162}$$

With some straightforward algebra:

$$\rho' = \frac{36(M - \delta M)}{25 + 15u_{0x} - 15u_{0y} - 9u_{0x}u_{0y} - 15u_{0x}^2 - 15u_{0y}^2}. \tag{163}$$

The condition corresponding to equation (149) for this geometry is:

$$\sum_{i \neq 1,2,3,7,8} f_i^{(1)} = 0. \tag{164}$$

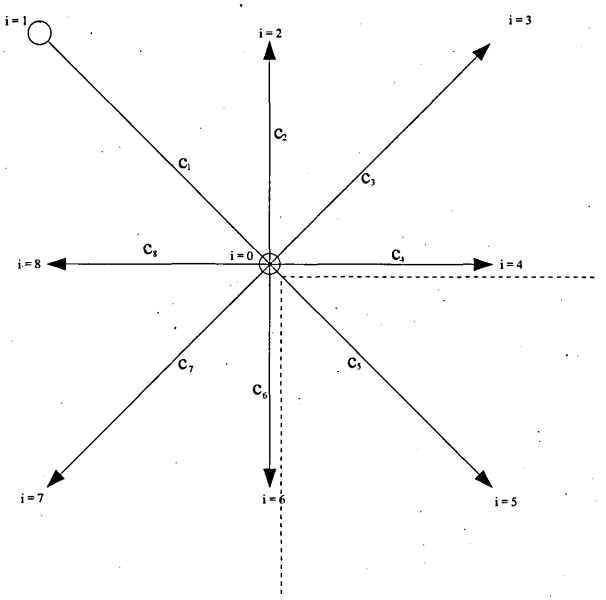

Figure 15: A boundary node on an external corner. The walls locations are denoted by the dashed lines; the node is taken to lie in the fluid. Propagated link data is denoted by a solid arrowhead or solid square on the link, unavailable link information (lack of data) by an open circle.

Solving for pre–collision $f_i^{(1)}$'s follows the procedure in section 6.4.1. No forbidden triplets of obtainable $f_i^{(1)}$'s arise for the case of 2D, D2Q9 internal corner geometry.

6.4.3 External Corner Algorithm

Figure 15 shows a top left external corner. The dotted line shows the supposed location of the boundary.

$f_0^\dagger$, $f_1^\dagger$, $f_2^\dagger$, $f_3^\dagger$, $f_4^\dagger$, $f_6^\dagger$, $f_7^\dagger$ and $f_8^\dagger$ are required. For M, δM and ρ' we now have:

$$M = \sum_{i \neq 1} f_i, \tag{165}$$

and:

$$\delta M = \sum_{i \neq 5} \phi_i. \tag{166}$$

Forbidden set of three free $f_i^{(1)}$'s	Link representation
$f_0^{(1)}$, $f_6^{(1)}$, $f_7^{(1)}$	
$f_0^{(1)}$, $f_4^{(1)}$, $f_3^{(1)}$	

Table 5: Diagrammatic representation of the forbidden combinations of $f_i^{(1)}$'s as free variables, for the external corner geometry of figure 15. Note that a large dot indicates inclusion of the rest link. Note also that, unlike table 4, only triplets which contain evaluatable $f_i^{(1)}$'s are listed here.

and (after some algebra):

$$\rho' = \frac{36(M - \delta M)}{35 - 3u_{0x} - 3u_{0y} + 3u_{0x}^2 + 9u_{0x}u_{0y} - 3u_{0y}^2}. \tag{167}$$

The mass conservation condition corresponding to equation (149) for this geometry is:

$$\sum_{i \neq 5} f_i^{(1)} = 0. \tag{168}$$

Solving for pre–collision $f_i^{(1)}$'s again follows the procedure in section 6.4.1. Triplets of evaluatable $f_i^{(1)}$'s now forbidden are listed in table 5.

Clearly other geometries and other situations are amenable to the boundary analysis of this section. Situations in 3D, corresponding to any number of cut links are a straightforward generalization of current method. Further extension to off–lattice boundaries is more complicated; it requires no fundamental modification to the method.

6.5 Results

In order to assess the above method for time-dependant boundary conditions, the development of incompressible flow in a uniform channel is considered, width W, bounded by stationary surfaces $y = constant$ and, in the horizontal direction, by

periodic boundary conditions. Flow was driven (for $t > 0$) by a uniform pressure gradient $\frac{\partial P}{\partial x} = g$ corresponding to $\phi_i = t_p G c_{ix}$ at $t = 0^+$ and the lattice fluid was initialized by assigning $f_i(\mathbf{r}, 0) = f_i^{(0)}(\rho, \mathbf{0})$ everywhere. Note that $G = 3g$ [28]. To generate additional interest the planar boundaries $y = 0^-, W^+$ were assumed to admit to slip, described by a slip velocity instantaneously in accord with equation (125). A range of slip lengths $b = 0.005, 0.05, 0.5$ were used. With no-slip boundary conditions, the corresponding flow has a characteristic time [46]:

$$T_0 = \frac{\nu}{W^2}. \tag{169}$$

v_s/V_0, the instantaneous slip velocity, normalized to the slip velocity measured at $t = T_0/2$, was obtained from a numerical solution (explicit, time-marching scheme) of the 2D, time-dependant Navier-Stokes equations, for times $t < T_0/2$; this solution is, in figure 16, found to be in excellent agreement with values obtained from the corresponding LB simulations using our boundary closure, over the range of slip lengths.

Consider now the steady state corresponding to the flow considered above, now with no-slip, Dirichlet boundary conditions. Invoking translational invariance in the x–direction and the fact that $\rho = \textit{constant}$ everywhere, it is straightforward to obtain the steady–state solution, $(v_0(y), 0)$, to the lattice fluid's Navier-Stokes equation (135):

$$v_0(y) = \frac{G}{6\rho\nu} y(W - y), \tag{170}$$

which we now proceed to use quantitatively to assess the accuracy of the solution obtained from simulations closed using the method of section 6.4.1.

Flow rate and hence Reynolds' number:

$$Re \equiv \frac{\overline{v}W}{\nu} = \frac{GW^3}{36\rho\nu^2}, \tag{171}$$

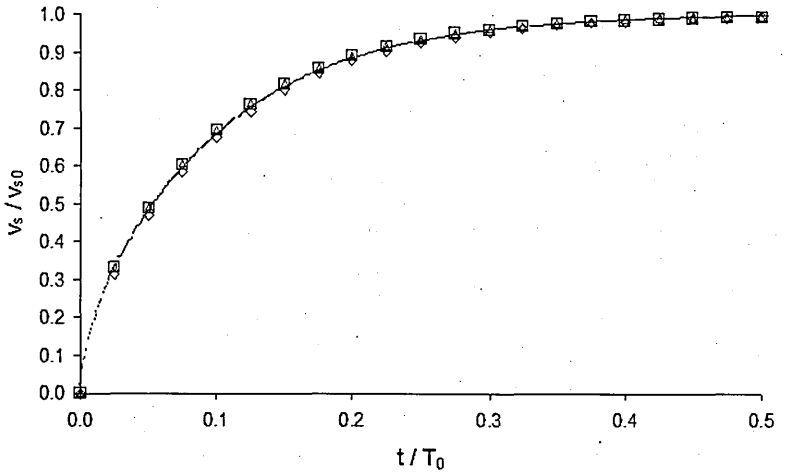

Figure 16: Evolution of the boundary slip velocity in pressure-driven duct flow. The time normalization parameter $T_0 = \frac{\nu}{W^2}$. The line shows the numerical solution, the open triangles correspond to slip length $b = 0.005$, squares $b = 0.05$ and diamonds $b = 0.5$.

was controlled by the value of G used in the source term $\phi_i = t_p G c_{ix}$ of the LBGK adapted evolution equation (126). A total absolute error:

$$\Delta_W \equiv \sum_{y \neq 0} |v_m(y) - v_0(y)| \,, \tag{172}$$

was defined, where $v_m(y)$ denotes the simulated, parabolic profile obtained with the boundary closure method of section 6.4.1. $v_m(y)$ was measured at constant $Re = 0.5$ for a range of lattice widths (resolutions) $W = 10 \times 2^N$, $N \leq 5$.

The data of figure 17 shows the value of $log_{10}(\Delta_W)$ as a function of resolution, measured by $\log_{10} W$. The collision parameter for this particular data was $\tau = 2/3$, however, use of $\tau = 1, 2$ produce no change in the data presented in figure 17 visible to the eye, as discussed below.

The results in figure 17 comprise two almost overlying, sets of points (a) and (b). Data set (a) is comprised of open squares and diagonal crosses, data set (b) of open circles and erect crosses. Sets (a) and (b) are based upon different choices of the

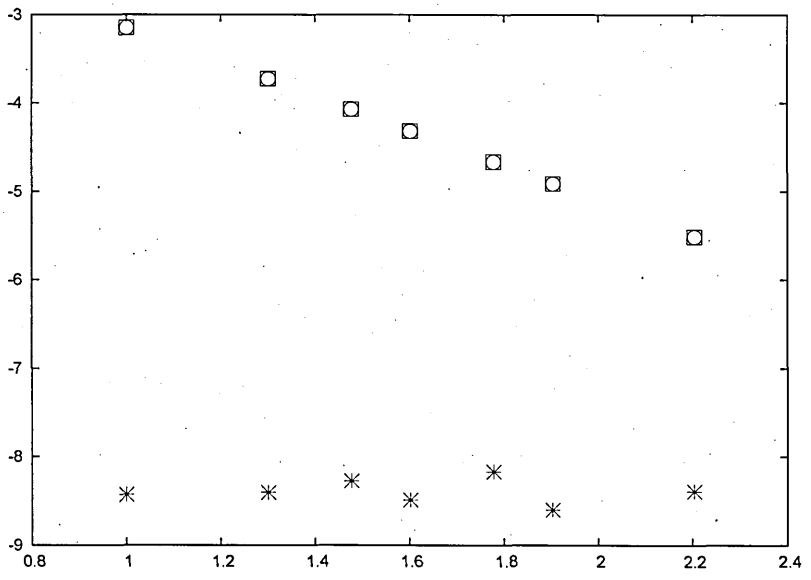

Figure 17: Error, ΔW, defined in equation (172), as a function of lattice resolution, W, for our method of section 6.4.1. This data was obtained for pressure driven flow in a uniform duct $y = 0, W$ with constant $Re = 0.5$. Collision parameter $\tau = 3/2$ for this data; values $\tau = 1, 2$ produce no change in the data visible to the eye. The results in this figure comprise two almost overlying, sets of points (a) and (b). Data sets (a) and (b) are based upon different choices of the free variable $f_i^{(1)}$'s; in terms of the planar boundary $y = W$ of section 6.4.1, set (a) corresponds to $i = 0, 1, 2$, set (b) to $i = 0, 3, 4$. In data set (a) open squares (diagonal crosses) represent unforced boundaries $\phi_i = 0, y = 0, W$ (forced boundaries $\phi_i \neq 0, y = 0, W$). In data set (b) open circles (erect crosses) represent unforced boundaries $\phi_i = 0, y = 0, W$ (forced boundaries $\phi_i \neq 0, y = 0, W$).

free variable $f_i^{(1)}$'s; in terms of the planar boundary $y = W$ of section 6.4.1, set (a) is characterised by choice $i = 0, 1, 2$, set (b) to choice $i = 0, 3, 4$. This point shall be discussed further below. In data set (a) open squares (diagonal crosses) represent unforced boundaries $\phi_i = 0, y = 0, W$ (forced boundaries $\phi_i \neq 0, y = 0, W$). In data set (b) open circles (erect crosses) represent unforced boundaries $\phi_i = 0, y = 0, W$ (forced boundaries $\phi_i \neq 0, y = 0, W$). In both sets (a) and (b), omitting the boundary force produces a linear variation in $\log_{10} W$ with $\log_{10} W$ of gradient ≈ -2, characteristic of a spatially second–order accurate scheme; in the case of forced boundaries $y = 0, W$, the method of section 6.4.1 appears to achieve a solution exact to machine precision, over the range of resolutions considered.

The fact that both data sets (a) and (b) in figure 17 correlate very well implies that the boundary closure method presented here is insensitive to the choice of free variable $f_i^{(1)}$'s. The results also appear to show that the method is stable over a range of values of collision parameter $2/3 < \tau < 2$.

Figure 18 shows corresponding reference data, obtained from various forms of bounce back boundary conditions. All the data in figure 18 were obtained with collision parameter $\tau = 1$, corresponding to a known optimum of performance in the bounce–back closure method. Before exposing any shortcomings, it should be noted that on–link bounce back is adaptable, robust, versatile and easy to implement for both LBGK and LB methods. That said, the limitations of the first-order accurate 'on–link' variant (represented by open squares) are clear. Applying a boundary force to this method (open circles) leaves the data unaffected. Mid-link bounce-back (diagonal crosses) approaches second order accuracy; with a boundary force (erect crosses) it achieves a smaller error, again approaching second order accuracy. None of the bounce-back methods recorded in figure 18 achieve the accuracy of the forced-boundary method of section of 6.4.1; neither can they be used to represent moving boundaries.

In order to demonstrate the ability of the method to handle both a boundary force

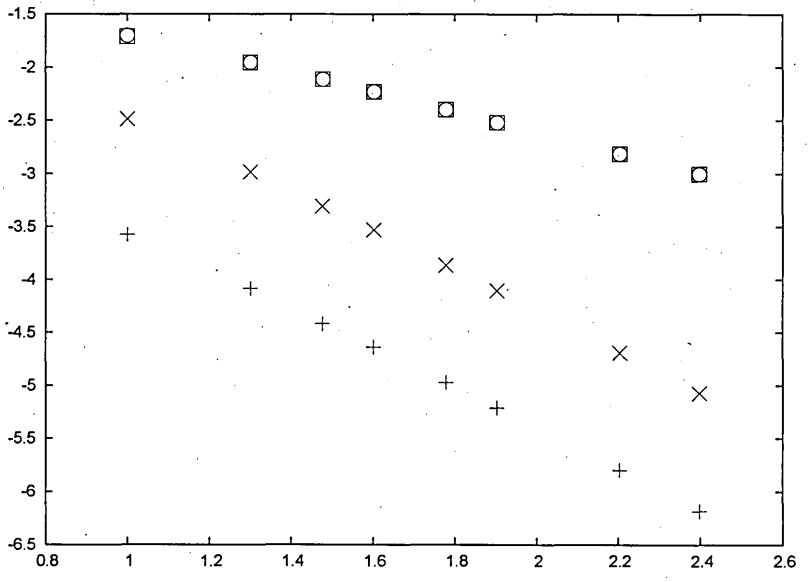

Figure 18: Error, ΔW, defined in equation (172), as a function of lattice resolution, W, for mid-link and nodal bounce-back boundary conditions. This data was again obtained for pressure driven flow in a uniform duct $y = 0, W$ with constant $Re = 0.5$. For all data in this figure, the collision parameter $\tau = 1$, corresponding to an optimum of performance for the bounce-back boundary method. Open circles (squares) correspond to nodal bounce-back (with a body-force applied at the boundaries $y = 0, W$). Diagonal (erect) crosses correspond to mid-link bounce-back (with a body-force applied at the boundaries $y = 0, W$).

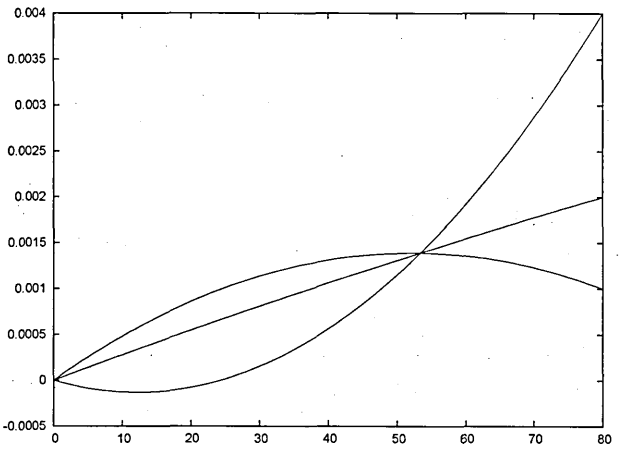

Figure 19: Sheared, pressure driven velocity profiles obtained from simulations using our boundary closure of section 6.4.1. For this data, $Re = 0.5$, collision parameter $\tau = 1$ and the right-hand wall velocity $u_0 = 1 \times 10^{-3}, 2 \times 10^{-3}, 3 \times 10^{-3}$, where all velocities are in LB units.

and a boundary velocity, sheared, duct flow was simulated. Shear and a pressure gradient were simultaneously applied to an incompressible fluid, again confined in a uniform channel of constant width $W = 80$, bounded by surfaces $y = 0, 80$. Flow was driven by (i) a pressure gradient, G, and (ii) a constant motion of the $y = W$ boundary, which moved at velocity $u_0\widehat{x}$. Various values u_0 and g consistent with a constant $Re = 0.5$ were used. For such unidirectional flow it is possible to superpose solutions of the Navier-Stokes equations:

$$v_0(y) = \frac{G}{6\rho\nu} y(W - y) + \frac{u_0}{W} y. \tag{173}$$

Figures 19 and 20 show a range of such velocity profiles with different values of u_0 and G. Figure 19 shows simulation data obtained for collision parameter $\tau = 1$ for $u_0 = 1 \times 10^{-3}, 2 \times 10^{-3}, 3 \times 10^{-3}$; figure 20 shows simulation data for $\tau = 2/3$ for $u_0 = 6 \times 10^{-3}, 7 \times 10^{-3}, 9 \times 10^{-3}$, where all velocities are in LB units. In all cases, any difference with the analytical result, obtained from equation (173), is too small to be visible in figures 19 and 20.

To asses the internal corner boundary closure of sub–section 6.4.2 a square, 2D

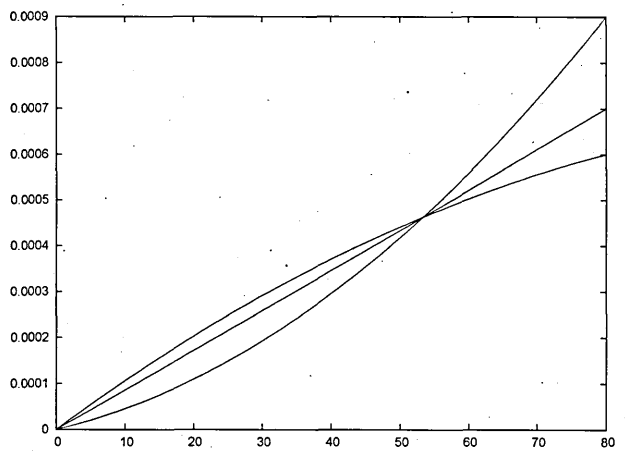

Figure 20: Sheared, pressure driven velocity profiles obtained from simulations using our boundary closure of section 6.4.1. For this data, $Re = 0.5$, collision parameter $\tau = 2/3$ and the right-hand wall velocity $u_0 = 6 \times 10^{-3}, 7 \times 10^{-3}, 9 \times 10^{-3}$, where all velocities are in LB units.

lid–driven cavity with rest boundaries $y = 0$, $x = 0, W$ and a moving lid $y = W$ was simulated. This lid is taken to move with velocity $\mathbf{u_0}$. Later $\mathbf{u_0} = u_0 \widehat{e}_x$ will be set. To obtain results which may validated in an original way and to illustrate the adaptability of our method, we introduce at this point a model variant. It is possible to simulate at $Re = 0$ by using an adapted D2Q9 LBGK scheme, after Ladd and Verberg, [47]. Following Ladd and Verberg, simplify the equilibrium distribution:

$$f_i^{(0)} = t_p \rho \left(1 + \frac{1}{c_s^2} \mathbf{v}.\mathbf{c}_i \right), \tag{174}$$

producing a D2Q9 LBGK model otherwise identical to that considered in sections 6.2 and 6.4.1. Analysing with the particular Chapman-Enskog approach of Hou *et al* in reference [27], it is possible to obtain a *steady-state* behaviour:

$$\frac{\partial}{\partial x_\beta} \rho v_\beta = 0, \qquad \frac{1}{6} (2\tau - 1) \nabla^2 \rho v_\beta - \frac{1}{3} \frac{\partial}{\partial x_\beta} \rho = 0, \tag{175}$$

which, with replacement $\mathbf{v} \rightarrow \rho \mathbf{v}$, represents the Stokes equations with pressure $P = 1/3\rho$ and shear viscosity $\eta = \frac{1}{6(2\tau-1))}$. The results of section 6.4 may be

modified for this revised model. For example, in section 6.4.1, equation (147) for the density is modified for use with the truncated equilibrium in equation (174):

$$\rho' = \frac{M - \delta M}{\sum_{i \neq 1,2,3} t_p \left[1 + \frac{\mathbf{u_0} \cdot \mathbf{c}_i}{c_s^2}\right]} = \frac{6M}{5 + 3u_{0y}}, \tag{176}$$

in which $\delta M = 0$, corresponding to no boundary force. Similar modifications must be made to the expressions for ρ' in the case of the internal corner, in equation (163):

$$\rho' = \frac{36(M - \delta M)}{25 + 24u_{0x} - 15u_{0y}}. \tag{177}$$

Using equation (174), the equilibrium, $f_i^{(0)}$, component of the boundary f_i's was assigned; boundary $f_i^{(1)}$'s were calculated and the post-collision distribution assembled precisely after the method of section 6.4.1.

Figure 21 shows detail of the rectangular stream function close to a corner of a, 2D, square lid–driven cavity at steady-state. Here the stream functions should approximate the known solution of flow into a corner [10], which appears in figure 22, for reference. The large system size of 150 lattice units and the small lid velocity 1.0×10^{-4} underlie encouraging correspondence with figure 22

6.6 Summary

In summary of this section a very adaptable and accurate method for closing a lattice Boltzmann LBGK simulation lattice has been set–out and validated. The method allows for both boundary motion and constant boundary fluid external forces such as gravity and its derivation demonstrates an accuracy consistent with the corresponding bulk scheme, it is instantaneously accurate and robust in that it functions with unimpaired accuracy for a range of values of LBGK collision parameters.

The boundary closure method has been developed for key flat and corner geometries; it may be applied to others. Overall it is suggested that this method is as

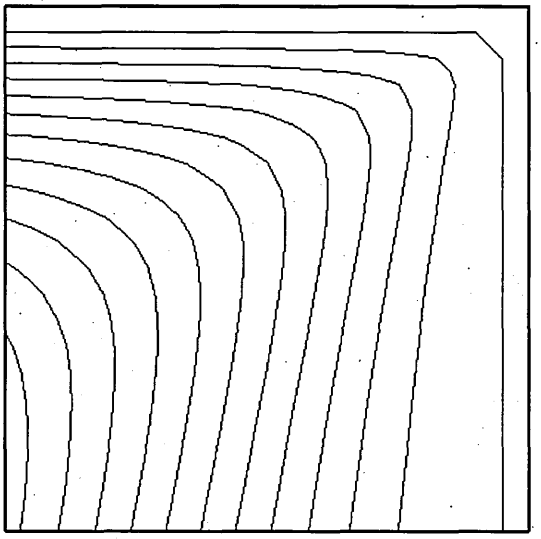

Figure 21: Corner Top, right hand corner detail of the rectangular stream function contours of a steady–state, 2D, square lid–driven cavity. This data was obtained for $Re = 0$ Stokes flow. The large system size of 150 lattice units and the small lid velocity underlie the good correspondence with the analytical solution of corner flow, shown in figure 22. $\psi = 0$ on $x = 0$, $y = 0$; the maximum value of ψ corresponds to $\psi = 1.05 \times 10^{-4}$.

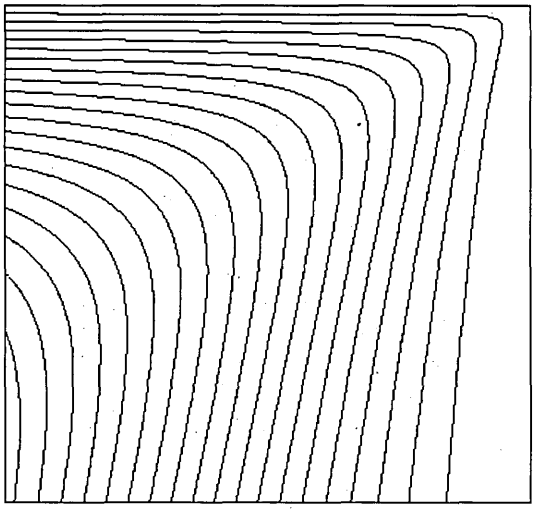

Figure 22: Rectangular stream function, ψ, contours for two dimensional flow into a top right, right-angled corner; $130 < x < 150$, $130 < y < 150$. Note, the image detail does not show the $\psi = 0$ on $x = 0$, $y = 0$; the maximum value of ψ corresponds to $\psi = 9.5 \times 10^{-5}$, in good agreement with the result in figure 21. Note the different contour interval in this image.

adaptable in this respect as popular bounce-back methods but, in addition, it has the added advantages of tractability, the capacity to handle both moving and stationary boundaries and the capacity to accommodate a fluid force at the boundary. The latter is of special significance to the wetting problem, to which this method will be applied later.

It is appropriate to emphasize that additional conceptual and therefore algorithmic complexity is introduced when the external force in question is not constant. The problem of a spatially-dependant body force arises when an LB fluid is subject to a "boundary force". We defer further consideration of issues relation to interfacial forces at a boundary until chapter 10, after spatially variable fluid body-forces have been considered in general in the next chapter.

7 Simulation at Low Reynolds and Capillary numbers for Multi–Component LB

In this chapter I begin to consider MCLB in detail. This chapter serves as a survey of the problems associated with MCLB in the continuum limit when addressed with the particular variant which I inherited. It is shown in this chapter that the problems identified may be reduced with minor algorithmic adjustments however for more complete resolution further fundamental modifications must be made to the method; these are discussed in following chapters, pradominantly chapter 8 and 9. That is not to say that the modifications and observations of this chapter are later rendered entirely redundant as they will be utilised in chapters 8 and 9 to help characterise the algorithmic modifications. At the risk of being repetitious, it is appropriate at this juncture to reiterate some LB introduction. This is done for completeness and with the aim of making the chapter relatively "self–contained".

7.1 Preliminary Remarks

The key issue revised in this chapter is to be found in section 7.5.1. In Chapter 4 we detailed the Chapman Enskog analysis by which the macroscopic dynamics of LB models are derived. However in 2002, Guo *et al* [48] showed that when a spatially variable external body force is applied to a fluid (as opposed to the constant body forces considered in Chapters 5 and 6) a more accurate varient of LB is possible. The changes necessary to implement the LB varient of Guo are straightforward (see section 7.5.1) however the process of modified Chapman Enskog expansion by which they are derived is intricate. This analysis is found to be in [48]. Guo's method as set out in 7.5.1 is retained throughout the rest of this thesis.

7.2 Background and Context

The simulation of multi–component fluids at low Reynolds number and low capillary number is of interest in a variety of applications such as the modelling of venule scale blood flow and microfluidics; however, such simulations are computationally demanding. In this chapter, the inherited multi–component lattice Boltzmann scheme is presented with some algorithmic modifications. The aim is to represent interfaces in the continuum approximation, the modifications presented are shown (i) significantly to reduce common algorithmic artifacts and (ii) to recover full Galilean invariance; see section 7.3. The method will be used to model drop dynamics in shear flow in two dimensions where it recovers correct results over a range of Reynolds and capillary number greater than that which may be addressed with the inherited method.

Many multi–component flows are conveniently modelled using the LB method; microfluidic multi–component flows are a relevant example. The latter are characterised as complex, incompressible flows at small Reynolds number, Re, and small capillary number, Ca. Multiple blood cells in venule–scale flows have also been represented with LB as immiscible drops (a traditional device) [49, 50]; such flows have $Re \sim 1$ and may be simulated using values of $Ca \sim 2.0$. Traditional computational fluid dynamics has also, of course, been applied in similar situations; recently Jadhav *et al* [51] have used the immersed boundary method to three dimensional simulations of leukocyte (white blood cell) rolling and, in the wider context, Esmaeeli and Tryggvason [52] and references therein, have published simulations of bubbles at low Re; and Sankaranarayanan *et al* [53] have produced a comparative study of LB and front-tracking finite-difference methods for bubble simulations.

Recall sections 4.3.1 and 4.3.3; there, a method due to Lishchuk was described which (i) produces sharp interfaces with small micro-currents (*ie* 'spurious velocities') [3]), (ii) has an independently adjustable interfacial tension (iii) is conve-

niently direct and (iv) can produce interfacial tensions larger than Gunstensen's method [24]. The key feature of the Lishchuk method is that, in order to impose surface tension, it impresses a curvature–dependant force in what is effectively a single fluid (this shall be revised later in chapter 11). However, both methods have a problem reaching low Ca and drop Reynolds number, Re_d (defined in sec 3.1), giving rise to a tendency for drops to "facet" as Ca and Re_d decrease (lengths of interface with zero curvature form parallel to lattice directions). Faceting is associated with a reduced drop advection and hence Galilean invariance (i.e. in all internal frames of reference, physical laws remain the same). The aim of this chapter is to show how the following issues may be addressed. The limited success we achieve motivates the developments of chapters 8 and 9.

We summerize the developments to be presented in the present chapter as follows:

In order to reduce discrete lattice effects and to enforce the global conservation of momentum, the interface method of Guo *et al* [48] is used in the implementation of interface forcing.

To attempt to reduce simulation noise and to increase accuracy, numerical derivatives of $o(4)$ (in mesh–spacing) are used to attempt to achieve a more distributed forcing around the sharp phase–field boundary.

Algorithmic modifications are made to enforce mutual impenetrability of binary fluids at the interface i.e. a kinematic condition. Which although partially achieved in this chapter, the methods proposed are somewhat primitive and computationally expensive. They are included for purposes of completeness rather than their algorithmic advances. A much more elegant and implementable solution is presented in Chapter 9.

7.3 Problems Associated with Reduced Reynolds and Capillary Number Simulations Using MCLB

The objective of the work reported in this chapter, to which the present section provides background, is to begin to modify and bound Lishchuk's method [25], to improve the representation of completely immiscible stiff drops in slow flow using, essentially, the MCLB algorithm I inherited, which is outlined in section 4.3. In essence the data in the present section serves to explore its faults and hence to set the agenda for following chapters.

Consider the simulation of the flow of high volume fraction, non–Brownian colloids in complex, internal geometry. At some positions in the geometry, the flow velocity is likely to be small. Now suppose we need to simulate stiff drops. We therefore require a small capillary number, Ca and must expect to encounter a small drop Reynolds' number, Re_d.

As discussed previously; LB fluids are described by the Navier–Stokes and continuity equations only for restricted Mach number $M \equiv U_0/c_s$, [3]; usually $M \leq 0.1$ is required. The fact that LB models have a small speed of sound, $c_s \sim o(1)$, limits the value of U_0. It follows that low Ca, low Re_d must be achieved using large surface tension, α and limiting the shear rate, $\dot{\gamma}$ see section 3.1. But a velocity which is too small is also a problem. In the interfacial region of multi–component LBs, there exist un–physical velocity field fluctuations, or micro-currents, generated by the interface algorithm. To obscure these, one approach to target Re_d and Ca is to limit length scale and maintain the characteristic velocity. Where length scales are limited, narrow interfaces are necessary to avoid introducing a length scale which may conflict with that of the flow problem *eg* . [49].

The above requirements translate to a need for (i) a sharp phase–field boundary, (ii) large surface tensions and (iii) a small LB micro-current. Whilst Lishchuk's method [25] facilitates (i)..(iii), as $Re \rightarrow 0$ a lack of Galilean invariance arises as

droplets 'pin' on the lattice.

Wagner [54] has pointed–out the origin of the microcurrents in multi–component LBs; imperfect correlation between an interface–producing force and the dynamics of the phase index, or colour label, ρ^N (see equation 91 and section 4.2.2) means fluid is kept in constant motion near the interface. Enhanced consistency between the interface–inducing force and the ρ^N field decreases the micro-current. These considerations maybe set in the context as follows.

By working from the core algorithm presented by Lishchuk in [25], the force applied to create the interface (discussed in Chapter 4 and further in section 9.2) is defined in terms of derivatives of ρ^N (see equation (182)). Sharp interfaces lead to numerical error in the calculation of these spatial gradients of ρ^N, resulting in anisotropic forcing over the a drop perimeter. Micro-currents and small, damped shape fluctuations ensue. By distributing the interface, the accuracy of numerical gradients may be improved, making the interface forcing more isotropic (see Chapter 8). Figure 23 shows data obtained from simulations which use an idealised phase field:

$$\rho^N(x,y) = \tanh\left(k\sqrt{(x-x_0)^2+(y-y_0)^2}\right), \tag{178}$$

in which the interface thickness parameter, k, has been varied. See the figure caption for simulation data.

With $k \approx 2.4$ (typical of numerical colour segregation) the interface width is less than two lattice spacings. Decreasing k distributes the interface and improves numerical derivatives of ρ^N (we shall return to interfacial distribution in detail in Chapter 8 and to a lesser extent in chapter 9). The interface–inducing force integrated along a radial section of the interface measures interfacial tension [25]. A numerical integral was obtained for cuts at (i) 0 rad and (ii) $\pi/4$ rad. R is the ratio of this force integral in (i) to that in (ii). In figure 23, as k decreases, $R \rightarrow 1$ and the total micro-current activity falls. The latter was measured as the lattice sum of the

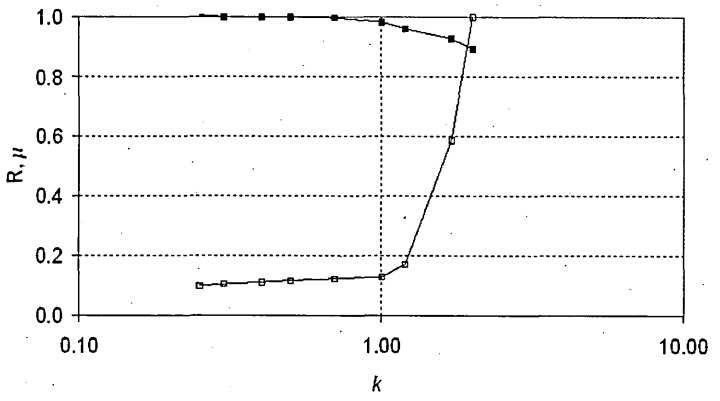

Figure 23: Data from the multi–component LB algorithm of reference [25]. Normalised, whole–lattice micro-current activity, μ, (open squares) and surface tension isotropy, R, (solid squares) for a range of interface thickness parameter, k. Ratio R is obtained as described in the section 7.3. This data was obtained for a neutrally buoyant drop, initial radius 15 lattice units, placed on a 75×75 lattice, with collision parameter $\tau = 1$.

velocity modulus, $\sqrt{u_x^2 + u_y^2}$, normalised to its value for $k = 2.4$. A more accurate interface–generating force appears to reduce micro-currents. In the present context, accuracy relates to the quality of numerical derivatives.

Next consider an immiscible drop embedded in a fluid, moving with uniform velocity U_0. In any multi–component LB the interfacial micro-current increases in proportion to surface tension, α. As U_0 decreases, the speed at which the drop advects, U_d, approaches the value which characterises micro-current activity and there is a loss of advection for the drop. This is discussed in more detail in section 7.8. Figure 24 shows the results of advection tests using the inherited algorithm of reference [25], outlined in section 7.4. A drop embedded in uniform flow is characterised by ordinal value $U_d/U_0 = 1$, a completely trapped drop by $U_d/U_0 = 0$. This data was compiled for a range of α and U_0. All data lie about the solid line inserted in figure 24, which is a grid search optimised fit to the data using an expression of the form $\tanh(a(U_d/U_0) - b)$. It was found that $a = 0.049$. From figure 24 it is clear that for $U_0/\alpha \leq 0.1$ there results a progressive loss of advection in the basic algorithm of

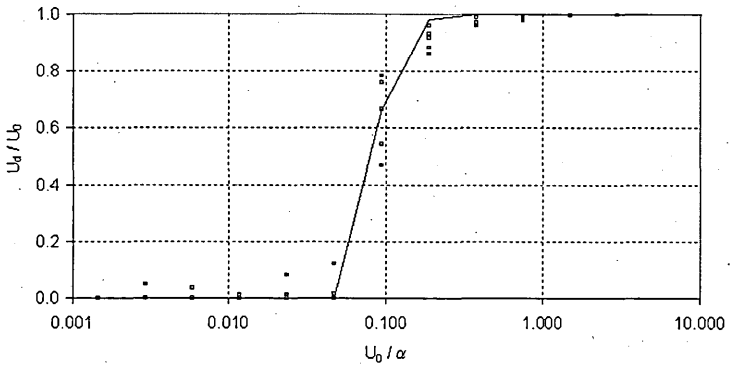

Figure 24: Original LB algorithm of reference [25]. The final advection speed, U_d, of an infinite chain of red drops, surface tension α, responding to 'far–field 'motion $(U_0, 0)$ is normalised to U_0 and plotted as a function of ratio U_0/α. This data was obtained for a neutrally buoyant drop, initial radius 15 lattice units, placed on a 75×75 lattice, for a range of values of boundary velocities, U_0 and interfacial tensions α. The solid line is a grid–search optimised tanh profile fit to the data. As the surface tension increases for fixed advection velocity, the drop 'stalls '.

reference [25]. Note that the results of figure 24 do not set a characteristic minimum on the value of quotient U_0/α; micro-current activity is, recall, slightly influenced by fluid viscosity.

The 'simple' case of a lack of Galilean invariance expressed in figure 24, is compounded in what we shall term a 'general' notion of Galilean invariance. Consider again the immiscible drop; its motion is a response to the accumulation of interfacial stresses. However, in all multi–component LB, interfaces are created by a perturbation which degrades the hydrodynamic stresses and, therefore, the drop's dynamics. Since this effect becomes more noticeable when stresses are relatively small, it is evident that we need a refined multi–component LB model first for the small Re, small Ca regime. In the present context, the benefits of the improved method arise in part because the interface perturbation better respects viscous stress. Results show general loss of Galilean invariance properties at low Re_d, low Ca, assessed by means of lift simulations, are improved by the innovations introduced in section 7.5 without compromising on the length–scale of the interface.

It is stressed that, for small Ca, large Re_d, the basic model outlined in the next section exhibits fewer problems. It is upon slow, $Re_d \sim 1$, surface–tension dominated, $Ca < 2$, flow that the reported innovations focus.

7.4 Multi–component Lattice Boltzmann in the Continuum Approximation

The work reported in this chapter is based upon the LBGK method, due to Qian *et al* [4], described previously. To this a 'source term', ϕ_i is added, allowing us to insert interfacial tension:

$$f_i(\mathbf{r} + \mathbf{c}_i, t+1) = f_i(\mathbf{r}, t) - \frac{1}{\tau}\left(f_i(\mathbf{r}, t) - f_i^0(\rho, \rho\mathbf{u})\right) + \phi_i. \tag{179}$$

Recalling Chapter 6, a constant source term, ϕ_i, Chapman–Enskog analysis [3] leads to modified Navier-Stokes equations: [28], [25]:

$$\frac{\partial}{\partial t}\rho u_\alpha + \frac{\partial}{\partial x_\beta}\rho u_\beta u_\alpha = -\frac{\partial}{\partial x_\alpha}\rho + \frac{\partial}{\partial x_\beta}\frac{\partial}{\partial x_\beta}\left(2\rho\nu S_{\alpha\beta}\right) + \mathbf{F}_\alpha, \tag{180}$$

where the source term ϕ_i is responsible for a constant body force:

$$\mathbf{F} = \sum_i \phi_i \mathbf{c}_i. \tag{181}$$

Note, the summation convention on repeated Greek subscripts applies in equation (180). For present purposes, the body force needs to contain spatial variation: $\mathbf{F} \rightarrow \mathbf{F}(\mathbf{r})$. This necessitates spatial variation in $\phi_i \rightarrow \phi_i(\mathbf{r})$ which, in turn, complicates the derivation of equation (180) above [48]. The solution given by Guo *et al* to this problem, which we utilise in section 7.5, requires a more complicated relationship between $\mathbf{F}(\mathbf{r})$ and $\phi_i(\mathbf{r})$ than that given in equation (181) and a re–definition of $\mathbf{u}$ [48].

The two fluids are segregated numerically as described in [24] and previously in section 4.3.1.

The quantity $\mathbf{F}(\mathbf{r})$ is defined in terms of the gradients of $\rho^N(\mathbf{r})$. To achieve a cross–interfacial pressure step proportional to the local curvature in ρ^N, K, we take:

$$\mathbf{F}(\mathbf{r}) \equiv \frac{1}{2}\alpha K \nabla \rho^N, \tag{182}$$

with the surface tension parameter α. Note that $\mathbf{F}(\mathbf{r})$ has also no explicit time dependence and that the right hand side of equation (182) vanishes for constant ρ^N. The cumulative effect of the force (in equation (182)) produces a pressure step across the interfacial region. Assuming K to be locally constant, the local pressure step is obtained as the line integral of $\mathbf{F}(\mathbf{r})$ between the terminal points, A and B, of a short, normal path:

$$\Delta P = \frac{1}{2}\alpha K \left[\rho^N(B) - \rho^N(A)\right] = \alpha K. \tag{183}$$

Note that A and B are embedded in the blue and red fluids respectively and so that $\rho^N(A) = -\rho^N(B) = 1$. Using the negative of a normalised colour gradient as the interface normal, $\widehat{\mathbf{n}}$, interface curvature K is obtained from the surface gradient:

$$K = n_x n_y \left(\frac{\partial}{\partial y} n_x + \frac{\partial}{\partial x} n_y\right) - n_x^2 \frac{\partial}{\partial y} n_y - n_y^2 \frac{\partial}{\partial x} n_x, \tag{184}$$

where $\widehat{\mathbf{n}} \equiv \nabla\rho^N / \left|\nabla\rho^N\right|$. It is important to note that the evaluation of K involves numerical second derivatives of ρ^N.

For the body–force defined in equation (182), arguments are presented in [25] which yield a source term, ϕ_i of the form:

$$\phi_i(\mathbf{r}) \equiv \frac{3}{2} t_p \alpha K \left(\nabla \rho^N\right) . \mathbf{c}_i, \tag{185}$$

which result, it is stressed, is improved in section 7.5.

In order to achieve narrow interfaces, numerical de-mixing of red and blue fluids at an interface is necessary [24,55]. The interface algorithm outlined in this chapter is based upon a variable force applied locally in a single fluid, to eliminate curvature in a phase field. This accurately represents interface dynamics (stress conditions) [25]. But the extent to which the kinematic condition of mutual impenetrability is implicit is unclear; the fact that there is a single 'sum' fluid means that the combined momentum of the mixture of fluids is continuous across an interfacial region but the velocities of the red and blue fluids may not be easily defined close to the interface.

7.5 Improvements to the Model

We describe a number of modifications to the algorithm outlined thus far. In summary:

1. The source term, ϕ_i, in equation (179) is modified more accurately to recover the required interface force in equations (180); this is achieved using the methods of Guo [48] and Ladd and Verberg [47].

2. ϕ_i is refined by more careful definition of the interface.

3. A higher–order accurate calculation of numerical derivatives in ϕ_i is used.

4. A kinematic condition is introduced.

7.5.1 Interface Source Term

A constant source term $\phi_i = 3t_p\mathbf{G}.\mathbf{c}_i$ is a well–known device for inserting a constant gradient $\mathbf{G}$. As stated in 7.4 it is recognised that spatial dependence in ϕ_i requires a modified Chapman–Enskog analysis of the macroscopic dynamics [28, 48]. The method of Guo is a progression of earlier work, in particular of Ladd and Verberg [47].

It may be used to yield an expression for a source term, ϕ_i, in terms of the target, space–dependant macroscopic force $\mathbf{F}(\mathbf{r})$ [48]:

$$\phi_i(\mathbf{r}) \equiv t_p \left(1 - \frac{1}{2\tau}\right) \left(3\left(\mathbf{c}_i - \mathbf{u}^*\right) + 9\left(\mathbf{c}_i.\mathbf{u}^*\right)\mathbf{c}_i\right).\mathbf{F}(\mathbf{r}), \tag{186}$$

where macroscopic velocities are now defined in terms of the external body force:

$$\mathbf{u}^* \equiv \frac{1}{\rho}\left(\sum_i f_i \mathbf{c}_i + \frac{1}{2}\mathbf{F}(\mathbf{r})\right). \tag{187}$$

Equation (186) replaces that given in equation (185); equation (187) gives an accompanying re–definition of the model's velocity. Note that equation (186) remains correct if the macroscopic force also contains time dependence. The Navier–Stokes equation (180) now acquires position dependence in the body–force term:

$$\frac{\partial}{\partial t}\rho u_\alpha^* + \frac{\partial}{\partial x_\beta}\rho u_\beta^* u_\alpha^* = -\frac{\partial}{\partial x_\alpha}\rho + \frac{\partial}{\partial x_\beta}\left(2\rho\nu S_{\alpha\beta}^*\right) + \mathbf{F}_\alpha(\mathbf{r}). \tag{188}$$

Note also the right hand side of equation (186) reduces to the right hand side of equation (185) for $\mathbf{u}^* \to 0$ and $\tau \to 1$.

The remainder of this subsection is concerned with peripheral comments on the preceeding method of Ladd and Verberg [47] which has been included for purposes of reference, but has no direct effect on work carried out here. The casual reader may wish to ommit the remainder of this subsection.

In addition to requiring a re–defined velocity, Guo's method removes the local relationship between the fluid strain rate and the distribution function, $f_i^{(1)}$ [48], see section 10.5, the usual form of which is reproduced in equation (204), section 7.8. For some applications this is unfortunate; Ladd points–out that local expressions for the strain rate are advantageous when calculating, for example, stresses imparted onto Lagrangian particles suspended in the LB fluid [56]. The approach of Ladd and Verberg [47] for time–independent body forces, applied to our case of an interface

force, yields:

$$\phi_i(\mathbf{r}) \equiv t_p \left(3 \left(\mathbf{c}_i - \left(1 - \frac{1}{2\tau} \right) \mathbf{u} \right) + 9 \left(1 - \frac{1}{2\tau} \right) (\mathbf{c}_i.\mathbf{u})\, \mathbf{c}_i \right) .\mathbf{F}(\mathbf{r}), \tag{189}$$

which recovers the Navier–Stokes equations (188) to the same accuracy as the method of Guo. However, the continuity equation now acquires an additional term [48]:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_\beta} \rho u_\beta = -\frac{1}{2} \frac{\partial}{\partial x_\beta} \mathbf{F}_\beta (\mathbf{r}) . \tag{190}$$

This approach, referred to in reference [48] as 'method 2a', requires no re–definition of velocity, and represents (in my opinion for narrow interfaces) an optimum treatment of variable forcing without recourse to a re–defined velocity. However, when using method 2a for a spatially varying force, the continuity equation acquires non–physical terms as indicated in equation (190). The latter may, in fairness, be small and in certain applications, an intact interpretation of velocity may outweigh this disadvantage. For present purposes however, method 2a is expected to produce inferior results owing to this continuity equation effect.

Equations (186) together with equation (187) (where appropriate) and equation (189) represent expressions for $\phi_i(\mathbf{r})$ each of which is an improvement upon our original expression in equation (185). We compare results from both methods in table 6; everywhere else we use equation (186).

7.5.2 Interface Definition and Cumulative Forcing

The criterion for inclusion of ϕ_i in the evolution equation (179) is modified to a condition on the gradient of ρ^N. ϕ_i is now only included if $|\nabla \rho^N| < 10^{-8}$. With $o(4)$ numerics (see section 7.5.3), this distributes the interface slightly forcing for a more accurate cumulative effect. Meanwhile, numerical segregation ensures that the length–scale of the variation in ρ_N is unaffected (the effects off using an algorithm with a larger degree of distribution is seen in Chapter 8). Choosing to use the

$1/\tau$	ν	Re	Re_d	Ca	y/L_y ± 0.01	U_d/U_0 ± 0.01	δy	$\delta y'$	$\delta y''$
0.85	0.23	1.11	0.10	2.98	0.49	0.49	0.45	1.92	1.93
0.90	0.20	1.23	0.11	2.70	0.48	0.48	0.78	1.72	1.72
0.95	0.18	1.36	0.12	2.43	0.49	0.48	0.61	1.54	1.54
1.00	0.17	1.50	0.16	2.21	0.51	0.51	0.49	1.12	1.12
1.05	0.15	1.66	0.15	1.99	0.49	0.48	0.43	1.35	1.35
1.10	0.14	1.83	0.17	1.80	0.50	0.50	0.59	1.91	1.81
1.15	0.12	2.03	0.18	1.63	0.50	0.50	0.19	3.25	5.20

Table 6: Summary of (i) the normalised steady–state transverse position y/L_y and (ii) the normalised steady–state velocity u_d/U_0 for neutrally buoyant drops, initial radius 15 lattice units, exposed to a uniform shear flow of characteristic shear velocity U_0.

internal (say) 'red ' relaxation parameter $\tau = \tau_R$ only when $\rho^N(\mathbf{r}, t) > 0.99$; this follows a 'level-set' philosophy and places the interface entirely within the embedding fluid. It has the effect of sharpening the phase field boundary and the location of any viscosity change.

Figure 25 shows the value of ρ_N versus radial distance for a static drop of initial radius 15 lattice units. The component index, ρ^N, changes essentially over a distance of one lattice unit. The solid line in figure 25 corresponds to the function $\tanh(k(r - r_0))$, with r_0 the average radius of the drop. Least–squares fitted values of $k = 2.77$ and $r_0 = 14.82$ were determined. The collapse of all interface data onto this single, contour supports the view that (i) whilst very narrow, the interface in our modified model has a structure independent of its orientation relative to the underlying lattice and (ii) to good approximation, the drop is isotropic. However from the slight spread of values it is apparent that further improvements may be attainable, see chapters 8 and Chapter 9.

Suppose the interface forms a sharp, closed contour in a plane. From the definition of curvature it can be shown that the line integral, along this contour, of a normally–directed force, proportional in magnitude to local contour curvature, K, must vanish. However, numerical derivatives introduce inaccuracies in K; a non-zero value of this line integral results. Therefore, the small, fluctuating but non–zero,

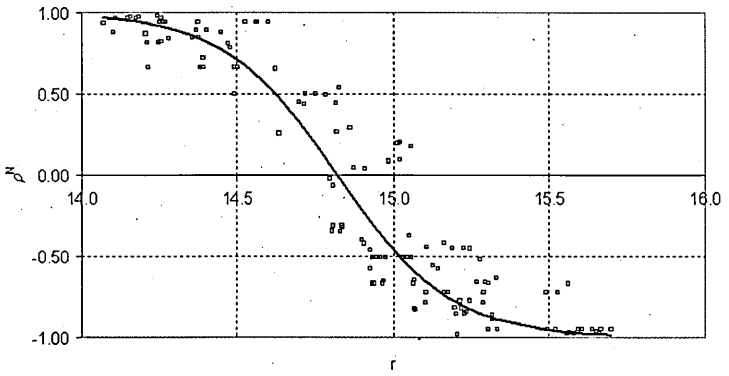

Figure 25: The distribution of phase field index for a neutrally buoyant red drop ($\rho^N = +1$) at rest, suspended in a blue fluid ($\rho^N = -1$). Both fluids have collision parameter $\tau = 1$ The initial radius of the drop is 15 lattice units, the interfacial region has a characteristic width of about a single lattice unit. The solid line corresponds to a tanh profile fit (see text).

total applied interface force is recorded and any imbalance is eliminated by applying its negative over the interface region, weighted by the magnitude of local interface force. The corresponding correction to the source term of equation (186) at position $\mathbf{r}$ is:

$$\delta\phi_i \equiv -3t_p \left(\frac{\sum_{\mathbf{r}'} \mathbf{F}(\mathbf{r}').\mathbf{c}_i}{\sum_{\mathbf{r}'} |\mathbf{F}(\mathbf{r}')|} \right) |\mathbf{F}(\mathbf{r})| \,, \tag{191}$$

where the summation on $\mathbf{r}'$ runs over all points in the interface region. Note that the factor '3' in (191) cancels when taking first moments, see (192).

To police the cumulative interface force it may seem reasonable to determine some form of local curvature average. However, sharp, local fluctuations in K are actually necessary to keep an interface sharp and smooth. Consider figure 26 in which interface sites with $-0.9 < \rho^N < 0.9$ have the direction of their colour gradient displayed. As the tangent to the local interface direction approaches a lattice direction local curvature K changes sign as follows. Point $(103, 82)$ in figure 26 lies in a region above a vertical length of interface, tangent to line $x = 106$. For a circular drop, the interface must, near to position $(103, 82)$, 'switch'and continue on another,

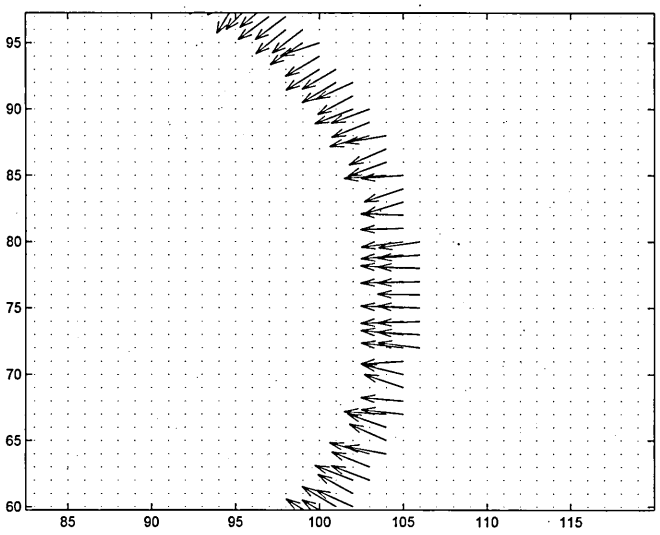

Figure 26: Detail of the normalised colour gradient field for a rest drop. Point $(103, 82)$ in the above figure lies in a region above a vertical length of interface tangent to line $x = 106$. In order to have an overall isotropic, circular drop the interface must 'switch 'to a parallel line of nodes. Near $(104, 82)$, the interface 'switches 'onto another, parallel line of nodes $x = 105$. When this happens the local curvature must fluctuate and change sign.

parallel line $x = 105$. When this happens the local curvature must fluctuate and even change sign. Local smoothing undermines this curvature elimination property of the algorithm, thereby undermining the mechanism by which the interface is kept smooth and sharp.

7.5.3 Calculation of Numerical Derivatives

The inter–node links $\mathbf{c}_i$, corresponding weights t_p, and indexing of the D2Q9 [4] lattice are defined in table 1. Note the implicit use of unit lattice spacing. By direct calculation this set of lattice link vectors and weights may be shown to have the

following even 'moments':

$$\begin{aligned} \sum_i t_p &= 1, \\ \sum_i t_p c_{i\alpha} c_{i\beta} &= \frac{1}{3}\delta_{\alpha\beta}, \\ \sum_i t_p c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} &= \frac{1}{9}\left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma}\right). \end{aligned} \tag{192}$$

Whilst the corresponding odd moments are zero. Now, a Taylor expansion of the function $\phi(\mathbf{r} + N\mathbf{c}_i)$, about $\mathbf{r}$, may be approximated:

$$\phi(\mathbf{r} + N\mathbf{c}_i) \simeq \phi(\mathbf{r}) + \sum_{n \leq n_0} \frac{1}{n!} N^n (\mathbf{c}_i . \nabla)^n \phi(\mathbf{r}), \tag{193}$$

where $N = 1, 2, 3 \ldots$. Retaining derivatives to $o(n_0)$ yields an expression with order of accuracy n_0 in lattice spacing. Multiplying equation (193) by the product $t_p\, c_{i\alpha}$ ($\alpha = x, y$), summing the result upon link index i and appealing to the moment properties in equations (192), one can obtain a system of simultaneous equations, with spatial derivatives such as $\partial^2\phi(\mathbf{r})/\partial x \partial y$ as its independent variables. The equations in this system are characterised by the choice of parameter α and the value of range parameter N. Taking $n_0 = 3$ and $N \leq 3$ one can solve this system for $o(n_0 + 1 = 4)$ accurate expressions for the first spatial derivatives of $\phi(\mathbf{r})$:

$$\nabla\phi(\mathbf{r}) = \sum_i t_p \left(\frac{9}{2}\phi(\mathbf{r} + \mathbf{c}_i) - \frac{9}{10}\phi(\mathbf{r} + 2\mathbf{c}_i) + \frac{1}{10}\phi(\mathbf{r} + 3\mathbf{c}_i) \right) \mathbf{c}_i. \tag{194}$$

Clearly this method of calculating gradients undermines that locality of the algorithm which is considered to be one of the strengths of LB. However, the additional overhead introduced only scales as the total length of interface in the simulation.

7.6 Kinematic condition

7.6.1 Kinematic Problem

It is worth noting at the outset of this sub–section that the following are initial attempts to enforce a kinematic condition and although successful to some degree, a more rigorous treatment of the kinematic condition is carried out in chapter 9 with superior results. The remainder of this section therefore should be mainly regarded as a record of the work I have undertaken, not as a worthwhile algorithmic extension.

The kinematic condition is stated as the requirement that, at the interface, the internal and external fluids have the same velocity. In fact the kinematic condition is important in multi–component flow. The $Re = 0$ calculation of flow past a spherical drop demonstrates this very clearly [12]. Here a method of coupling the dynamics of the phase field and those of the fluid are presented, aiming to embed a kinematic condition. Throughout sub–sections 7.6.1 and 7.6.2 an asterisk superscripts denotes discrete positions; point (x^*, y^*) is a discrete lattice position; (x, y) is a sub–lattice point.

The narrow interfaces generated by the methods considered here correspond, typically, to local interface profiles with only one node on the lattice with $|\rho^N(x^*, y^*, t^*)| < 0.8$. Take sub–lattice contour $\rho^N(x, y) = 0$ to define the (off-lattice) centre of the interface, suppose that all the N (M) points $\{(x_i^*, y_i); i = 1..N\}$ $(\{(x_i, y_i^*); i = 1..M\})$ at which this contour intersects the short lattice links have been determined (by searching vertically and horizontally over the lattice and using linear interpolation). Contour $\rho^N(x, y) = 0$ bounds what should be two different fluid domains; boundary and, ideally, kinematic conditions should apply between the bulk fluids on this contour. $\{(x_i^*, y_i); i = 1..N\} \bigcup \{(x_i, y_i^*); i = 1..M\})$ is a suitable set of points upon which to apply appropriate conditions. Of course, this choice of a set of points implicitly defines a sharp interface.

7.6.2 Kinematic Condition Solutions

Noting that the dynamic boundary conditions on interfacial stress are implicit in the core inherited method [25], and aiming to preserve resolution and to conserve local momentum, we consider three practical approaches to a kinematic condition, each based upon the set of points $\{(x_i^*, y_i); i = 1..N\} \bigcup \{(x_i, y_i^*); i = 1..M\})$.

Method 1 Take the two sets of pairs of points $\{((x_i^*, int(y_i) + 1), (x_i^*, int(y_i) - 1)); i = 1..N\}$ and $\{((int(x_i) + 1, y_i^*), (int(x_i) - 1, y_i^*)); i = 1..M\}$, where $int(x_i)$ denotes the integer part of x_i. Find the mean velocity $\overline{\mathbf{u}^*}$ of each such pair of points:

$$\overline{\mathbf{u}^*} = \frac{\rho(int(x_i) + 1, y_i^*)\mathbf{u}(int(x_i) + 1, y_i^*) + \rho(int(x_i) - 1, y_i^*)\mathbf{u}(int(x_i) - 1, y_i^*)}{\rho(int(x_i) + 1, y_i^*) + \rho(int(x_i) - 1, y_i^*)}, \tag{195}$$

and assign the pre–collision fluid velocity at points $(x_i^*, int(y_i)+1)$ and $(x_i^*, int(y_i)-1)$ to $\overline{\mathbf{u}^*}$; it is velocity $\overline{\mathbf{u}^*}$ which is used to evaluate the equilibrium distribution $f_i^{(0)}(\rho, \overline{\mathbf{u}^*})$ in equation (179). Fluid just inside each of the separated components is thus induced to move at the same speed. It is important to note that it is the modified velocity, defined in equation (187), which is averaged.

Method 2 Determine the mean, pre–collision velocity $\overline{\mathbf{u}^{*\prime}}$ in an interfacial neighbourhood $(int(x_i) - 1) \leq x^* \leq (int(x_i) + 1), (int(y_i) - 1) \leq y^* \leq (int(y_i) - 1)$:

$$\overline{\mathbf{u}^{*\prime}} = \frac{\sum_{i=-1,0,1,} \sum_{j=-1,0,1} \rho(x^* + i, y^* + j)\mathbf{u}(x^* + i, y^* + j)}{\sum_{i=-1,0,1,} \sum_{j=-1,0,1} \rho(x^* + i, y^* + j)}. \tag{196}$$

As in method 1, all nodes in this region are then evolved according to equation (179) with an with equilibrium distribution $f^{(0)}(\rho, \overline{\mathbf{u}^{*\prime}})$. Clearly this method results in a loss of resolution in the interface region relative to method 1.

Method 3 Using an alternative, non–numerical method for component segregation [57, 58] covered extensively in chapter 8, with 'segregetion parameter' $\beta = 0.7$ (notation of reference [58]) method 1 from above was followed. Method 3 therefore has a distributed interface of much greater width than those obtained by the

numerical segregation otherwise considered; with $\beta = 0.7$ the value of ρ_N switches between -1 and $+1$ in the surface normal direction over approximately 6 lattice spacings (rather than 1 lattice spacings with numerical segregation shown in figure 25). One of the advantages identified at this stage of a diffuse phase boundary is that the contour $\rho^N(x, y) = 0$ is better defined. At this point the segragation algorithm variant of [57, 58] is introduced solely to obtain a contextual view of the kinematic condition it shall be exploited further in the following chapters.

7.6.3 Comments

There is a considerable body of literature relating to the transverse migration or 'lift'of a solid particle subject to simple shear or parabolic flow. Lift is widely discussed in the context of a range of important applications such as re–suspension. The term lift may suggest a direction of motion dictated by a gravitational field; however it is important to note that the effect is simply the migration of a particle or droplet in a flow as a consequence of a hydrodynamic interaction of the particle with one or more neighbouring boundaries. The direction of migration is determined by the nature of the flow and the boundaries. As a basis for evaluating the above model modifications I consider the lift of a deformable, incompressible, neutrally buoyant liquid drop in simple shear flow, at low Re_d, with a capillary number $Ca \sim 2$ which permits only restricted deformation.

The chosen geometry is a stringent test of the algorithm. For one thing, anticipating the results, lift appears to rely upon a small imbalance of hydrostatic pressure. To achieve this the interface algorithm must be sensitive. Secondly, for a small shear rate, a drop's drift velocity component in the direction of the unperturbed shear gradient is small, particularly as the drop approaches its steady–state position. Any lack of Galilean invariance is thus exposed in the transverse component of the motion.

7.7 Results

7.7.1 Simulations of drop lift in shear flow

It is important to use accurate boundary conditions. The necessary Dirichlet boundary conditions cannot be represented by popular LB methods like on–link or mid–link bounce back [3]. To impose $o(2)$ accurate velocity boundary conditions, a precursor method to that reported in chapters 5 and 6 was used. The precursor method used is the inherited boundary method reported in reference [46]. Recall by $o(2)$ accurate we mean (i) second–order accurate in lattice spacing *and* (ii) able correctly to construct boundary distributions f_i up to the level denoted $f_i^{(2)}$ [3].

Advection and lift tests identical to those described in section 7.3 were repeated using all four innovations described in section 7.5 and as previously mentioned, using an $o(2)$ accurate lattice closure scheme to impose velocity boundary conditions. In particular, the kinematic condition reported in method 1 of section 7.6.2 was used. In terms of the results in figure 24, simple Galilean invariance is restored down to values of parameter $U_0/\alpha \sim 0.003$. This modest improvement (compared with the data of figure 24) is to be expected from the arguments set–out in section 7.8.

Lift tests were conducted on a drop of initial radius 15 lattice units, placed on a $L_x \times L_y = 150 \times 51$ lattice, bounded by Dirichlet velocity conditions, reported in ref [46], in the y-direction and periodic boundary conditions in the x-direction. This, of course, produced an infinite line of drops. The drop was positioned with its centre at $y = 20$ lattice units and equilibrated for $5 \times 10^3 \leq t$ time steps. For time steps in the range $t:\ 5 \times 10^3 \leq t \leq 4 \times 10^5$ the $y = \{L_y, 0\}$ boundary was set to move with x–velocity $\{5 \times 10^{-3}, 0\}$ lattice units per time step respectively. What would, for a uniform system, be a shear profile $u_x = (U_0/L_y)y$ was thus applied to stable drops.

The centre of mass y–coordinate of the drop is denoted by y_0 and the x and y components of its velocity by u_d and v_d, respectively. Mechanical equilibrium or

steady–state is defined by a vanishing long–time average $< v_d >_t = 0$. At steady–state, general arguments require the centre of mass (x_0, y_0) of a neutrally buoyant drop to advect at velocity $(U_d, V_d) = (U_0/2, 0)$ at y position $L_y/2$. Results for $(< u_d >_t, < v_t >_t)$ from the simulation can therefore be compared with expected values $(U_0/2, 0)$.

For all the results in figures 27 - 30, and table 6, simulation parameters were $\rho = 2$, $\dot{\gamma} = 10^{-4}$, $\alpha \sim 3.4 \times 10^{-3}$. All data presented here assumes that the drop liquid and the embedding fluid have the same viscosity. The single LB relaxation parameter τ was varied over a range $0.85 \leq 1/\tau \leq 1.15$. Steady–state data is presented for $0.100 \leq Re_d \leq 0.165$ and $1.63 \leq Ca \leq 2.98$. The *channel* Reynolds number ranges in the data $1 \leq Re \leq 2$ (see table 6 for exact values).

Figures 27 and 28 are an abbreviated life history of a lifting drop. Figure 27 shows the partially deformed and orientated drop at the point at which transverse migration commenced, depicted using pressure $P \equiv \rho/3$ (for our model). Note the high pressure region visible between the drop and the closest, rest boundary. From these and similar images it is apparent that substantial drop deformation and orientation occur before any lift commences. Figure 28 shows the corresponding, now symmetric, pressure distribution about the drop at steady state. Both figures 27 and 28 show fluctuations in pressure in the interfacial region; the phase field, ρ^N, by contrast, varies slightly more smoothly across the interface (figure 25). However, as mentioned previously, the slight spread in data points indicates that further improvements may be possible. Further advancements are detailed in following chapters.

More data is shown in table 6 and figure 29. Values of y_0/L_y and normalised final advection velocities U_d/U_0 are recorded in table 6. The last three columns of table 6 compare δy, the absolute difference $|y_0 - L_y/2|$ with $\delta y'$, the corresponding data obtained with the method of Ladd and Verberg [47] (equation (189)) and $\delta y''$ that obtained using the unmodified inherited algorithm [25] (equation (185)).

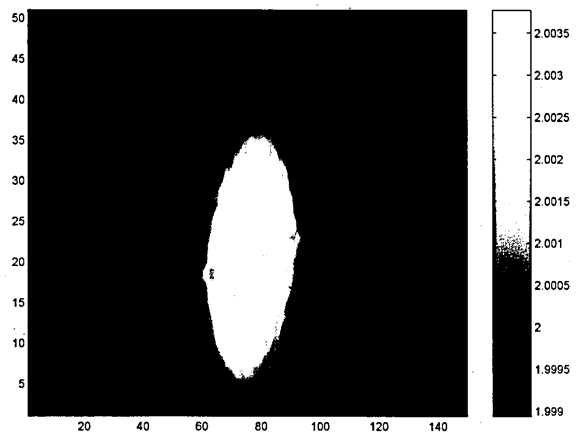

Figure 27: The pressure distribution for a droplet at the point at which transverse migration commences. Note the fluctuation in the pressure near to the interface.

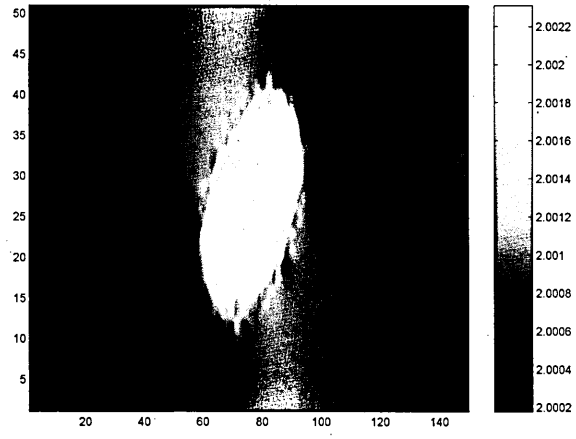

Figure 28: The pressure distribution for the same droplet as that presented in figure 27, at steady state. Note the small regions of high pressure located at the tips. Some slight faceting is visible in this image.

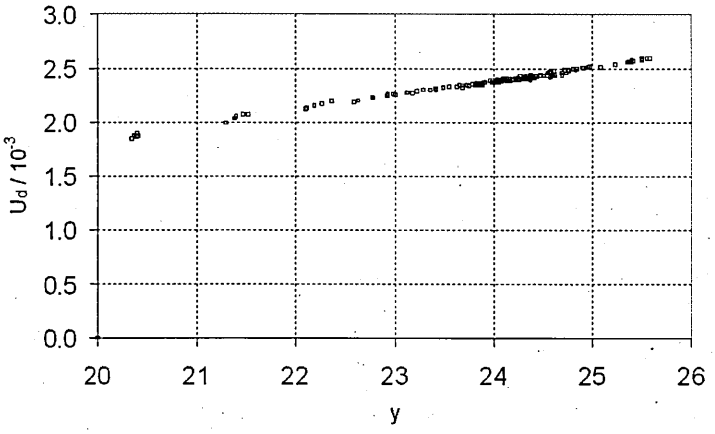

Figure 29: y–positions and x–velocities for all the trajectories of all the drops in table 6. y–excursion is plotted against the corresponding x–velocity, sampled every 2.0×10^5 time steps. Point $y = 25, U_d = 2.5 \times 10^{-3}$ is the equilibrium position required by general Galilean invariance.

Notwithstanding a need to adjust the velocity, the method based upon Guo's analysis [48] outperforms the others, as shown in typical drop centre of mass trajectories in figure 29. Note, however, that the Ladd–Verberg approach is not at its best in this application as it does not achieve the same accuracy as the method of Guo [48]. Its unadjusted velocity may well represent an advantage in other applications.

Figure 30 shows data for all the trajectories of all the drops in table 6. Their y–excursion is plotted against the corresponding x–velocity, sampled every 2.0×10^5 time steps. Point $y = 25, U_d = 2.5 \times 10^{-3}$ is the equilibrium position required by general Galilean invariance. All data lie on the same trajectory.

A measure of the Gallilean invariance of the current algorithm is the steady–state value of distance from mid–channel, δy. Outside the above range of parameter it begins to fail. However, an improved performance in terms of consistency is clear from a comparison of the last two columns of table 6. From the sixth row, for $Ca/Re \sim 1$, $Ca/Re_d \sim 10$, the modified model has 28% of the error of the unmodified inherited model of reference [25]; over the range of data in table 6, the error in the current (unmodified) algorithm, measured by δy ($\delta y''$), is 0.51 (2.10). With $\delta y \leq 0.5$ as

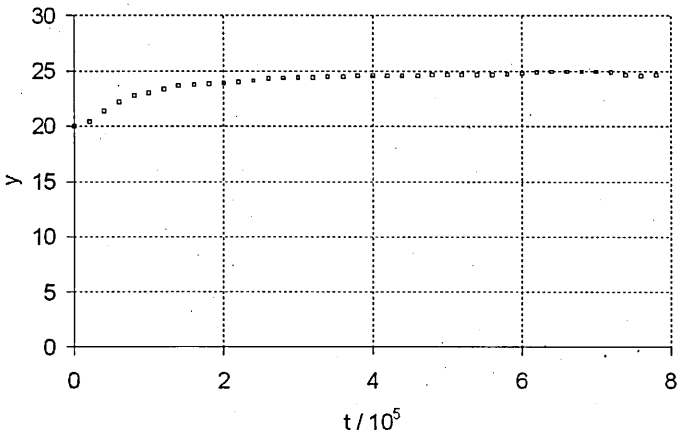

Figure 30: Centre of mass drop trajectory. Note that all the drops in the study appear to have a trajectory broadly indistinguishable from that shown here. For times in excess of the 7.0e5 time steps the drop remains at the channel centre position $y = 25$. Note also that a degree of pinning is apparent as the drop approaches its steady–state position.

the criterion, $Ca/Re_d \sim 0.78$ are possible with the modified algorithm. At no point does the unmodified algorithm have so small an error (associated value of $\delta y'$).

7.7.2 Kinematic condition

To illustrate the kinematic condition outlined in sub–section 7.6.1, and to set the context of the more successful and elegant methods of chapter 9, we proceed as follows. Consider the relative normal velocity in the region of the interface. At steady–state, the deformation of a sheared drop is constant. The component of fluid velocity relative to the drop in the direction of the interfacial normal, u_n, must vanish and the drop interface should correspond to a closed contour $u_n = 0$. The interface normal is readily obtained from the (negative of) the direction of $\nabla\rho^N$. The constant centre–of–mass velocity, $\mathbf{U_d}$, of a drop was calculated and subtracted from the steady–state velocity field to allow u_n to be calculated:

$$u_n = (u_x - U_{dx}).\left(\frac{-\frac{\partial\rho^N}{\partial x}}{|\nabla\rho^N|}\right) + (u_y - U_{dy}).\left(\frac{-\frac{\partial\rho^N}{\partial y}}{|\nabla\rho^N|}\right). \qquad (197)$$

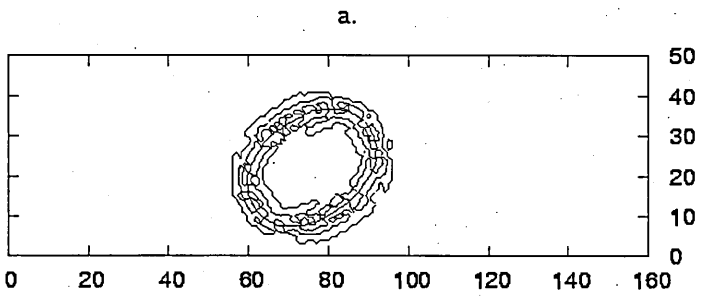

Figure 31: $Ca = 2.5$, $Re_d = 0.13$. Data for numerical segregation with (a) kinematic conditions of method 1 (b) kinematic condition of method 2 and (c) no kinematic condition (all (a)..(c) produce results with no differences visible to the eye). The fluctuating solid contour corresponds to zero surface normal velocity, $u_n = 0$ (see equation (25)). The interface, $\rho^N = 0$, corresponds to the axis of the 'toroidal' region covered by this fluctuating solid contour.

Note that $U_d/U_0 = 0.5$ in the channel centre, on general grounds.

Figures 31 and 32 each show the (solid) $u_n = 0$ contour. Both images correspond to $Ca = 2.5$, $Re_d = 0.13$. Figure 31 corresponds to the three cases of (a) no kinematic condition, (b) the kinematic condition of method 1 and (c) the kinematic condition method 2; figure 32 corresponds to method 3.

For no kinematic condition (figure 31), the fact that the $u_n = 0$ contour is so fragmented, relative to the $\rho^N = 0$ contour, demonstrates that, despite a sharp phase field boundary, the actual hydrodynamic boundary is diffuse. It was hoped that a kinematic condition would produce better correlation between the $u_n = 0$ and the interface. Unfortunately data corresponding to methods 1 and 2 show no improvement observable to the eye. Figure 32, which, recall, corresponds to a model extension by way of a diffuse interface, shows a large improvement both in terms of an identifiable $u_n = 0$ contour and its correlation with the contour $\rho^N = 0$. The diffuse interface is considered in detail in chapters 8 and 9. In respect of the diffuse nature of the interface, however, it is important to note that the noise in the $u_n = 0$ contour, clear in figure 31 extends over approximately six lattice sites in the normal direction; in this sense the hydrodynamic boundary between the fluids

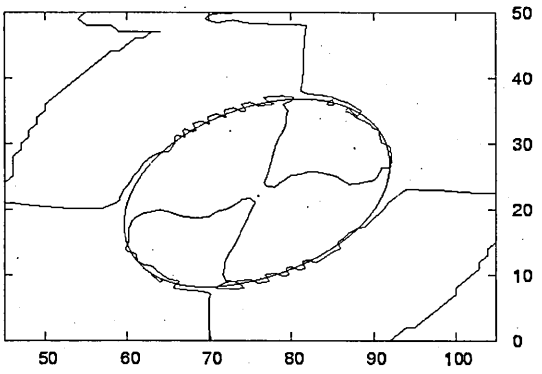

Figure 32: Kinematic condition of method 3, section 7.6.1 used. $Ca = 2.5$, $Re_d = 0.13$. Overlaid contours of $\rho^N = 0$ (closed ellipsoid) and $u_n = 0$. The correlation between the phase field and hydrodynamics information in this simulation is clearly much improved. Note that the two diagonal lines extending from (92,0) to (105,14) and (45,24) to (70,50) are artifacts of the way in which this information was compiled.

are approximately as diffuse (but rather more noisy) as the phase field boundary generated by the modified segregation shown in figure 32.

The fact that the modified segregation of figure 32 shows a $u_n = 0$ contour which is not closed, for example, at the poles of the drop is at this stage an undesirable error. Around the poles, flow is directed along the tangent to the interface and the normal component of velocity is very small throughout finite polar regions. Hence, with the modified segregation of method 3, section 7.6.1, the correlation between hydrodynamic boundary and phase boundary $\rho^N = 0$ is improved over the majority of the interface, however, small deviations still persist at the poles. This deviation is reduced in later chapters with the introduction of a diffuse interface and a more effective kinematic condition.

7.8 Discussion

In this section I intend to discuss issues relating to the inherited LB simulation model, reserving for remarks on the lift till section 7.9. Unless otherwise stated, suppose that the original model of [25] is in use.

Eventual loss of simple Galilean invariance seems universal in multi–component LB on the following approximate argument. Consider a two dimensional $D2Q9$ LB fluid. Suppose the lattice fluid is translating uniformly at velocity $\mathbf{U_0} = U_0\widehat{e_x}$. It is then described everywhere by an equilibrium distribution function $f_i^{(0)}(\rho, \mathbf{U_0})$; *eg.* equation (39). The momentum distribution for a link j, parallel with imposed flow, is thus shifted about its rest equilibrium value (of $t_p\rho$) by an amount:

$$df_j = 3t_p\rho U_0 + O(U_0^2). \tag{198}$$

Note, the numerical factor 3 in equation (198) has dimensions which give the overall expression for df_j units of ρ. Now, suppose the LB fluid contains an embedded drop, radius R. When the interface–inducing source term ϕ_j (equations (179) and (185)), competes with df_j, information relating to the advection of the interface is lost. This leads to an approximate, $O(U_0^2)$ accurate, condition for loss of flow information in the interface region:

$$3t_p\rho U_0 \geq \frac{3}{2}\frac{\alpha}{R}t_p\frac{\partial}{\partial x}\rho^N, \qquad R \equiv \frac{1}{K}, \tag{199}$$

where, recall, the factor $\frac{3}{2}$ is dimensional. Taking a value for the phase field gradient of unity (typical of the centre of the interface) and $R = 15$, we obtain:

$$\frac{U_0}{\alpha} \geq \frac{1}{2}\frac{1}{\rho R} = \frac{1}{2 \times 15 \times 2.0} \approx 0.035, \tag{200}$$

where we have substituted the value of $\rho = 2.0$ used for all the results.

Given the approximate nature of these arguments, the prediction in equation (200) is very well supported by the simulation results shown in figure 24; a fit to the data (solid line, figure 24) gives a figure of 0.049. Note that, using the modified forcing, defined in equation (186) or, indeed an alternative form of LB interface generating algorithm, the central assumptions of the above analysis survive.

Similar arguments apply to the problem of a drop subject to fluid stresses. The

lift simulations reported in the section 7.7.1 consider a drop suspended in a uniform applied shear:

$$u_\alpha = e_{\alpha\beta}x_\beta, \qquad e_{11} = e_{21} = e_{22} = 0, \qquad e_{12} = \dot{\gamma}, \tag{201}$$

which we take to be the unperturbed flow. There is an exact solution for the f_i's in this flow [59], for our model it may be written [60]:

$$f_i = f_i^{(0)}(\rho, \mathbf{u}) - \tau t_p \frac{\rho e_{\alpha\beta} Q_{\alpha\beta}}{b c_s^2}, \qquad Q_{\alpha\beta} \equiv \left(c_{i\alpha}c_{i\beta} - \frac{1}{3}\delta_{\alpha\beta} \right). \tag{202}$$

It is worth noting in passing that our unit time step, introduced from equation (179), actually multiplies the second term in the right hand side of this equation (see the Chapman–Enskog derivation [3] of Hou *et al* [27]) giving that term the overall dimensions of density. For our underlying D2Q9 model, the number of velocities $b = 9$ and $c_s^2 = 1/3$ (in units of lattice velocity squared) so equation (202) implies an expression for the so-called $f^{(1)}$ component (see *eg* [3] and Hou *et al* [27]) of the momentum distribution function:

$$f_i^{(1)} = -t_p \frac{3\tau\rho}{2} \dot{\gamma} c_{ix} c_{iy}, \tag{203}$$

in which, note, the factor 3 has units which cancel those of the factor $c_{ix}c_{iy}$, again giving the right hand side of equation (203) dimensions of ρ overall. Now, it is well–known that the strain–rate in single–component LB fluids is given by $S_{\alpha\beta} = -\frac{3}{2\rho\tau}\sum_i f_i^{(1)} c_{i\alpha}c_{i\beta}$ from which it is possible to show that the viscous stress is controlled by the $f_i^{(1)}$'s:

$$\sigma'_{\alpha\beta} = 2\eta S_{\alpha\beta} = \left(\frac{1}{2\tau} - 1 \right) \sum_i f_i^{(1)} c_{i\alpha}c_{i\beta}, \tag{204}$$

and that the LBGK evolution equation (179) can, with the assumptions of the

original model of reference [25], be written in terms of the $f_i^{(1)}$'s [3]:

$$f_i\left(\mathbf{r}+\mathbf{c}_i, t+1\right) = f_i^0(r,t) + \left(1-\frac{1}{\tau}\right) f_i^{(1)} + \phi_i. \tag{205}$$

Suppose now that the LB fluid contains a drop, radius R, under shear. It is the fluid stresses in the interfacial region which impress force between the drop and embedding fluid. Clearly, from equation (204), this stress is determined by the $f_i^{(1)}$'s. So if the interface–inducing source term ϕ_i competes with $f_i^{(1)}$, information relating to the fluid stress distribution about the interface is degraded. Hence the loss of stress information, or lift, is characterised by:

$$\phi_i \approx f_i^{(1)}. \tag{206}$$

Substituting from equations (185) and (203) into equation (206), cancelling the (dimensional) numerical factors and t_p, we find a simple condition for degradation of surface stress information:

$$\frac{\rho\tau\dot{\gamma}R}{\alpha\nabla\rho^N} \approx O(1). \tag{207}$$

Thus, taking the parameters used for the data of table 6, $\tau \sim 1$, $\rho = 2$ and $\dot{\gamma} = 10^{-4}$ and taking $\left|\nabla\rho^N\right| \sim 1$, equation (207) suggests an upper limit of surface tension $\alpha \sim 3 \times 10^{-3}$. This value accords with observations. In table 6, recorded values of $\delta y'$ show that the drops with a value of $\alpha = 2.5 \times 10^{-3}$ used there are not forced into the position required by general Galilean invariance. However, the results presented in table 6 for δy, using the improved model (for the same parameters) reflect a simulation with considerably improved general Galilean invariance. This is due, in part, to the use of the analysis of Guo *et al* [48], which ensures that the background fluid stress information is less corrupted by the interface forcing.

7.9 Conclusion

Presented in this chapter was a two–component lattice Boltzmann model of flow, designed to be efficient in the continuum approximation and to support essentially unstructured interfaces. It contains a set of modifications to my inherited model, of reference [25], described at the beginning of section 7.5. In outline, these modifications achieve a slightly improved fluid–fluid interface behaviour in terms of drop shape (faceting) at smaller capillarity number, Ca, and Reynolds numbers Re and Re_d. These improvements in the basic reliability of this class of LB model make it more suitable for microfluidic and non-Brownian colloid applications than the previous method, *eg* those in references [49,50], in which local Re_d variations must occur. The approximate analysis of section 7.8 suggests that the modifications outlined in this chapter achieve better results by improving the representation of stress near the interface.

We have used the improved model to report on 2D simulations of lift on slightly deforming, neutrally buoyant, immiscible liquid drops in simple shear. Clearly, on general grounds, a drop must migrate across stream lines of the undisturbed simple shear flow, until it reaches a position of mechanical steady–state: which is what was observed to a greater extent than with the previous inherited method. To use LB accurately one must be confident that any small pressure fluctuations can register stresses in the interfacial region of the fluids, where pressure was seen to be fluctuating due to interfacial forcing; a point underscored in figures 27 - 28, in which the relative size of the lift–inducing pressure difference and the interfacial pressure jump are made clear.

It is worth commenting that results also demonstrate that transverse position and drop velocity increase in proportion; also a marked degree of deformation and orientation in local flow precedes any lateral migration. This, again, is an unsurprising observation given uniform nature of applied shear. It is also worth repeating that

the migration trajectory information shown in figures 29 and 30 were not observed significantly to vary over all drops in this study.

Although improvements have been made, very small values of Ca and Re_d remain elusive with the method as it now stands. This regime might be achieved by reducing local shear rates $\dot{\gamma}$. From equation (207), this can be balanced only by a reduction in the value of phase field gradient $\nabla\rho^N$. Taken with the results of figures 31 and 32, this observation calls into question the practical utility of a sharp phase boundary close to which the velocity, in particular, fluctuates. The results in figure 32 are important in this respect. Figure 32 depicts a distributed interface algorithm but the velocity field correlation with a particular value of phase field is greatly improved- actually making the hydrodynamic boundary sharper, the distributed interface method shall be investigated fully in the following chapters.

7.10 Chapter Summary

The work in this chapter is of limited value, although not fruitless; it is worth noting some remaining shortfalls and where substantial advances in performance remain elusive.

As the capillary number tends to zero, pinning still occurs. Although the advances in this chapter have enabled lower Raynolds numbers and Capillary numbers to be reached, pinning remains at very low Ca. The cause of this pinning is thought to originate in the relatively high levels of noise that are still present in the phase–field that themselves are attributed to the "sharpness" of the interface causing high gradients. With the sharpness of the interface being one of the attractive attributes of this method; to increase the thickness of the interface requires a more complete solution of the kinematic condition.

The attempts to formulate a kinematic condition in this chapter have proven to be computationally expensive and only mildly effective. The incorporation of a

kinematic condition, however, is still considered to be of importance and shall be considered much more successfully in Chapter 9.

8 New Approach to Continuum MCLB Simulation

Close investigation of the phase field and other hydrodynamic signals using numerical re–colour revealed that a sharp contour was only achieved in the phase field with high levels of noise and corruption in other hydrodynamic quantities. As mentioned previously the large levels of noise present in other hydrodynamic quantities are mainly due to the large gradients in the colour field itself leading to noise associated with its derivatives. A proposed solution to this corruption is to attempt to achieve a smoother variation in the phase field coupled with an adequate means of identifying the fluid–fluid interface such as not to compromise interface resolution. That is, make the interface too broad for a continuum description to be appropriate.

8.1 Preliminary Remarks

Presented in this chapter is a multi–component lattice Boltzmann simulation for continuum fluid mechanics, paying particular attention to the component segregation part of the underlying algorithm. The principal result of this chapter is that the dynamics of a component index, or phase field, is, for the first time, obtained for a segregation method after d'Ortona *et al* [57], due to Latva-Kokko and Rothman [6]. The said dynamics accord with a simulation designed to address multi–component flow in the continuum approximation and underwrite improved simulation performance in two main ways: (i) by reducing the interfacial micro-current activity considerably and (ii) by facilitating simulational access to regimes of flow with low capillary number and drop Reynolds number, see Chapter 7. The component segregation method studied, used in conjunction with Lishchuk's method [25], produces an interface which is distributed in terms of its component index; however the hydrodynamic boundary conditions which emerge are shown to support the notion of

a sharp, unstructured, continuum interface.

8.2 Introduction

As identified previously, the simulation of multi–component fluids at low Reynolds number and low capillary number is of interest in a variety of important applications. Refining MCLB simulation to the extent that it can address this regime more effectively than the MCLB variant of Chapter 7 is the major objective here.

A variety of multi–component flows on a range of length and time scales have been modelled using the multi–component LB method; the work presented in this chapter is of most relevance to such applications, for instance the recent microfluidic flow simulations of Dupin *et al* [23] would be significantly improved in physical accuracy were they performed using the methods outlined in this chapter. Such flows as are addressed in reference [23] are formally characterised as complex, incompressible flows at small Reynolds number, Re, and small capillary number, Ca. Multiple blood cells in venule–scale flows have also been represented in this regime, with LB [49, 50]. Improving simulations of such non–Brownian colloids, addressed in the continuum fluid approximation, where fluid-fluid interfaces have no assumed structure and appear as boundary conditions, are the object of the work undertaken in this chapter.

Despite some successes [23] and work carried out in Chapter 7, continuum multi–component LB (with Lishchuk's method or Gunstnsen's method) still encounters a significant problem reaching low Ca and drop Reynolds number, Re_d; as outlined previously; as Ca and Re_d both decrease there is an increasing tendency for suspended drops of immiscible fluid to facet and to attach, or *pin* to the simulation lattice in an effective loss of Galilean invariance; here this problem is addressed together with the related problem of the interfacial micro-current, see Chapter 7.

In this chapter, in continuation of work carried out in Chapter 7, I draw attention

to what I regard as the key element of the multi–component LB algorithm which will impact on its uptake; an elegant, efficient alternative fluid-fluid segregation process, devised by d'Ortona *et al* [57]. Recently, Latva-Kokko and Rothman have demonstrated the advantages of a segregation method closely related to d'Ortona's in respect of an almost eliminated tendency to drop pinning [6]. Concentrating on application to the continuum regime, where drop evaporation must be completely interrupted, what is shown here is that, by adopting the approach of d'Ortona and Latva-Kokko (i) the interfacial micro-current is, likewise, almost eliminated and (ii) the kinematics of the interface may be predicted (see below). What emerges is a simple, relatively efficient, analytic algorithm with determinate phase-field dynamics, has very low micro-current activity, very low pinning and high adaptability in the continuum approximation.

8.3 Interfacial Hydrodynamics in the Continuum approximation

The key result of this chapter - one might say this thesis - is the equation which expressed the dynamics and the kinematics of the LB models interface. We will; in this chapter, seek the latter as the equation of motion of ρ^N, the phase field. In the continuum regime, the interface is defined as $\partial\Sigma_{12}$, the boundary between two Navier-Stokes domains Σ_1 and Σ_2. On this boundary certain kinematic and dynamic conditions must be satisfied. The most fundamental is the kinematic condition of mutual component impenetrability which requires that both components have the same velocity on the $\partial\Sigma_{12}$. [5]:

$$\underline{v}^{(1)} = \underline{v}^{(2)}, \quad on \quad \partial\Sigma_{12}. \tag{208}$$

The interfacial dynamics appropriate to the continuum regime have been discussed in 3.4 and [5].

To recover the dynamics expressed in (208) we would like to see the interface, identified by contours of ρ^N=constant, advect in flow without any form of diffusion, put another way, for a continuum interface our models phase field should obey;

$$\frac{D}{Dt}\rho^N = 0. \tag{209}$$

8.4 Multi–component lattice Boltzmann in the continuum approximation

The source term has been discussed previously in Chapter 6 and Chapter 7. Since the aim of this chapter is to generalise its application, some key details are presented with the aim of setting this chapter into context and for the sake of completeness and coherence. If the reader is happy with the material of Chapter 7 he or she may wish to proceed directly to section 8.5.

$$f_i\left(\mathbf{r}+\mathbf{c}_i, t+1\right) = f_i\left(\mathbf{r}, t\right) - \omega\left(f_i\left(\mathbf{r}, t\right) - f_i^{(0)}(\rho, \rho\mathbf{u})\right) + \phi_i(\mathbf{r}). \tag{210}$$

The source term, ϕ_i, in equation (210) has the effect of impressing a body force in the fluid which emerges from this kinetic (f_i-based) description; $0 \leq \omega \leq 2$ is the single selectable parameter which, with LBGK, controls the viscosity of the fluid [3]; all other symbols have their usual meaning. Below I discuss how the source term, ϕ_i, inserts a body force of limited range to produce an interfacial pressure step in the fluid. Note, equation (210) assumes unit time step.

For a constant source term, ϕ_i, Chapman–Enskog analysis [3] may be used to derive LB's characteristic weakly compressible form of the incompressible Navier-Stokes equations, now with a body force. For example, the choice:

$$\phi_i = t_p \frac{1}{k_2}\mathbf{F}.\mathbf{c}_i, \tag{211}$$

represents one, simple, widely used device for inserting a uniform body force (or uniform pressure gradient), $\mathbf{F}$, into the lattice Navier-Stokes equations:

$$\frac{\partial}{\partial t}\rho u_\alpha + \frac{\partial}{\partial x_\beta}\rho u_\beta u_\alpha = -\frac{\partial}{\partial x_\alpha}\rho + \frac{\partial}{\partial x_\beta}\left(2\rho\nu S_{\alpha\beta}\right) + F_\alpha, \tag{212}$$

where kinematic viscosity $\nu \equiv \frac{1}{6}\left(\frac{2}{\omega} - 1\right)$ and $S_{\alpha\beta}$ is the strain rate tensor; the lattice–dependant constant k_2 and the link weights, t_p, are discussed in Chapter 4 and re-iterated in the beginning of section 8.5.

The precise relationship between the fluid body force and the source term ϕ_i is an issue throughout this chapter. Recall, a constant microscopic source term, ϕ_i, in equation (210), and the resulting macroscopic body force, in equation (212), are related:

$$\mathbf{F} = k_2 \sum_i \phi_i \mathbf{c}_i. \tag{213}$$

For present purposes, however, the external body force (used to impress interfacial behaviour) has spatial variation: $\mathbf{F} \rightarrow \mathbf{F}(\mathbf{r})$ see Chapter 7, which necessitates spatial variation in $\phi_i(\mathbf{r})$; as Guo *et al* point-out [48], such a generalization complicates the derivation of equation (212) above and requires (i) a more complicated relationship between $\mathbf{F}(\mathbf{r})$ and $\phi_i(\mathbf{r})$ than that given for equation (213) and (ii) a re–definition of $\mathbf{u}$ (initially defined in Chapter 7). Before reiterating Guo's modified relationship, it will be useful to consider the form of fluid interface force which ϕ_i should generate, and where this force should be applied.

Fluid-fluid interface dynamics are applied in regions of the lattice where two immiscible fluids interact (and segregate). The two fluids concerned are, as in Chapter 7, designated red and blue. As different fluids mix in an interfacial region they define a single mixture, or sum, fluid which is evolved according to the evolution equation (210), the interface dynamics being captured by the source term, ϕ. The mixture is then segregated. Segregation is not a passive process; the way in which it is achieved influences the physical accuracy of the model. Where previously, as

described in detail in Chapter 7, numerical segregation has been used, here an analysis of a particular, advantageous, segregation method (initially used in Chapter 7 in subsection 7.6.2 method 3) is presented in section 8.5; in the remainder of this section, salient detail of the interface force and corresponding source term will be considered.

To identify mixed fluid, first distinguish between individual lattice fluids themselves. As stated previously our multi–component LB uses a phase field based upon the densities of 'red 'and 'blue' fluids present at a node. Following the notation of [25], we define a fluid component index, or phase field, $\rho^N(\mathbf{r})$:

$$\rho^N(\mathbf{r},t) \equiv \left(\frac{R(\mathbf{r},t) - B(\mathbf{r},t)}{R(\mathbf{r},t) + B(\mathbf{r},t)}\right), \qquad -1 \leq \rho^N(\mathbf{r}) \leq 1. \tag{214}$$

Red and blue fluids mix under the LB propagation step [3] defining an interfacial region; here an additional force is applied to the sum fluid capturing the effects of interfacial tension. The two fluids are then segregated numerically [24] or by the method of section 8.6. Whatever the segregation algorithm used, in Lishchuk's LB interface method [25] a surface tension inducing interface force, $\mathbf{F}(\mathbf{r})$, is defined in terms of the gradient of $\rho^N(\mathbf{r})$. To achieve a cross–interfacial pressure step proportional only to K, the local curvature in the ρ^N field, use a fluid body force:

$$\mathbf{F}(\mathbf{r}) \equiv -\frac{1}{2}\alpha K \nabla \rho^N, \tag{215}$$

with α, a surface tension parameter. Note, the right hand side of equation (215) vanishes for constant ρ^N. As stated previously, the negative of a normalised phase field gradient is used as the interface normal, $\widehat{\mathbf{n}}$, K is obtained from the surface gradient:

$$K = n_x n_y \left(\frac{\partial}{\partial y} n_x + \frac{\partial}{\partial x} n_y\right) - n_x^2 \frac{\partial}{\partial y} n_y - n_y^2 \frac{\partial}{\partial x} n_x, \tag{216}$$

where from Chapter 7:

$$\widehat{\mathbf{n}} \equiv -\frac{\mathbf{f}'}{|\mathbf{f}'|}, \qquad \mathbf{f}' \equiv \nabla\rho^N. \tag{217}$$

It is appropriate to emphasize here that all the gradients are calculated numerically using suitable finite differences, see Chapter 7; a source of significant computational overhead. We shall consider the feasibility of appropriate, *locally calculated* alternatives in section 8.9 and in section 8.10.

The cumulative effect of the force in equation (215) is to produce a pressure step across the interfacial region. Assuming K to vary slowly, the total local force perpendicular to a unit length of interface, which is numerically equal the pressure step, is obtained from the integral of the force (density), $\mathbf{F}(\mathbf{r})$, between the terminal points, P_R and P_B, of a short, normal path with element dn:

$$\Delta P = \left| \int_{P_B}^{P_R} \mathbf{F} dn \right| = \frac{1}{2}\alpha K \left[\rho^N(P_R) - \rho^N(P_B)\right] = \alpha K, \tag{218}$$

where P_B and P_R are embedded in the blue and red fluids respectively, so $\rho^N(P_R) = -\rho^N(P_B) = 1$.

where P_B and P_R are embedded in the blue and red fluids respectively, so $\rho^N(P_R) = -\rho^N(P_B) = 1$.

temp The interface algorithm outlined in this chapter is based upon a spatially and ture in a phase field. This, it may be shown, accurately represents interface dynamics (stress conditions) [25]. However, the extent to which the kinematic condition of mutual impenetrability is implicit is unclear; the fact that there is a single 'sum' fluid means that the combined momentum of the mixture of fluids is continuous across an interfacial region but the velocities of the red and blue fluids may not be easily defined close to the interface. Previous attempts at enforcing an interfacial kinematic condition, set out in Chapter 7 are, at least, clumsy and only partially effective. I return to this point in section 8.5 and in Chapter 9.

The methodology of Guo addresses the issue of a spatially varying body force in a progression of earlier work of, in particular, Verberg and Ladd [47]; it furnishes

an expression for an appropriate source term, ϕ_i, in terms of *variable* macroscopic force $\mathbf{F}(\mathbf{r})$ [48]:

$$\phi_i(\mathbf{r}) \equiv t_p \left(1 - \frac{\omega}{2}\right) \left(3\left(\mathbf{c}_i - \mathbf{u}^*\right) + 9\left(\mathbf{c}_i.\mathbf{u}^*\right)\mathbf{c}_i\right).\mathbf{F}(\mathbf{r}), \tag{219}$$

where, it is emphasised, the fluid velocity is re-defined, effectively to carry some of the influence of any external body force:

$$\mathbf{u}^* \equiv \frac{1}{\rho}\left(\sum_i f_i \mathbf{c}_i + (1-f)\mathbf{F}(\mathbf{r})\right), \qquad f = \frac{1}{2}. \tag{220}$$

I shall return to this point and the definition of link weights, t_p, in section 8.5. Note that equation (219) accounts for the case of a macroscopic force which also contains time dependence. Of course, in a standard LBGK model, $f = 1$ in definition (220). It is worth the inconvenience of retaining the parameter f, introduced above, as it is possible easily to adapt our key results to a standard LBGK model simply by changing its value from $f = \frac{1}{2}$ to $f = 1$.

With Guo's methodology [48], the model's Navier–Stokes equation (212) now acquires position dependence in the body–force term:

$$\frac{\partial}{\partial t}\rho u_\alpha^* + \frac{\partial}{\partial x_\beta}\rho u_\beta^* u_\alpha^* = -\frac{\partial}{\partial x_\alpha}\left(c_s^2\rho\right) + \frac{\partial}{\partial x_\beta}\left(2\rho\nu S_{\alpha\beta}^*\right) + F_\alpha(\mathbf{r}). \tag{221}$$

We emphasize that $F_\alpha(\mathbf{r})$ should be regarded as that interface force defined in equation (215).

8.5 Analysis of the Formulaic Segregation Rule

Previously, in Chapter 7, results derived from my inherited MCLB, based on the numerical segregation method of Gunstensen and Rothman [24], outlined in section 4.3.1 were shown. In this chapter we consider in detail the formulaic segregation

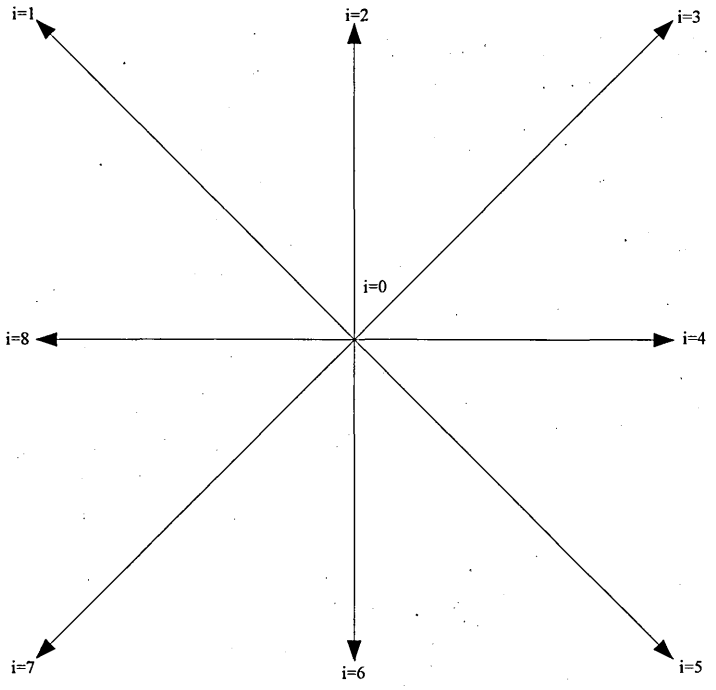

Figure 33: The nodal velocity set and subscripting convention for the two dimensional, nine velocity, D2Q9, lattice. Note that links classified with odd values of subscript i have a larger length. The weights corresponding to the links are listed in Table 1 of section 4.1.1.

method of Latva-Kokko and Rothman [58], briefly discussed in section 4.3.2.

Recall, R (B) denotes the density of red (blue) fluid present at the node, position $\mathbf{r}$. In terms of phase-field parameter, ρ^N, these quantities are:

$$R(\mathbf{r},t) = \frac{1}{2}\rho(\mathbf{r},t)\left(1+\rho^N(\mathbf{r},t)\right), \tag{222}$$

$$B(\mathbf{r},t) = \frac{1}{2}\rho(\mathbf{r},t)\left(1-\rho^N(\mathbf{r},t)\right). \tag{223}$$

For compatibility with previous chapters, simplicity and reasons of available resource, work in this chapter shall be carried out on the D2Q9 lattice defined in figure 33. Extensions to alternative lattice structures shall be made where appropriate. Results will generalize to D3QN transparently, for lattice and model dimensionality enter only through parameter k_2, which is retained explicitly throughout this

section. As previously a unit time step and lattice spacing are used for the same reasons. Whether in two or three dimensions, all lattices are considered have the usual isotropy properties:

$$\sum_i t_p c_{i\alpha} c_{i\beta} = k_2 \delta_{\alpha\beta}, \tag{224}$$

$$\sum_i t_p c_{i\alpha} c_{i\beta} c_{i\gamma} c_{i\delta} = k_4 \left(\delta_{\alpha\beta}\delta_{\gamma\delta} + \delta_{\alpha\gamma}\delta_{\beta\delta} + \delta_{\alpha\delta}\delta_{\beta\gamma} \right), \tag{225}$$

odd moments of the lattice basis vectors, $\mathbf{c}_i$, being zero. In equations (224) and (225) the link weights, t_p, are a set of scalar quantities which weight the directions of the lattice according to the length of velocity vector $\mathbf{c}_i$. The values of t_p and $\mathbf{c}_i$ for the of the D2Q9 LBGK lattice are defined in Table 1 of section 4.1.1. Note that, for the D2Q9 lattice $k_2 = \frac{1}{3}$.

Component segregation is achieved by allocating colour density R (say) to the sum fluid's momentum distribution function, f_i, optimally at each node [24]. It is the purpose of this chapter to demonstrate the advantages of, and to analyse, an alternative approach, after d'Ortona [57], due to Latva-Kokko and Rothman [6]. I shall adhere to the following convention. A post-collision, pre-propagation momentum density is indicated by the use of a dagger superscript; accordingly equation (210) becomes:

$$f_i^{\dagger}(\mathbf{r}, t) = f_i(\mathbf{r}, t) - \omega \left(f_i(\mathbf{r}, t) - f_i^{(0)}(\rho, \rho\mathbf{u}^*) \right) + \phi_i(\mathbf{r}), \tag{226}$$

in which the modified velocity of (220) is used to calculate the equilibrium, note. Note also that the post collision momentum distribution function, $f_i^{\dagger}$, includes the source contribution, ϕ_i, defined in equation (219).

For convenience, the collision process is subsequently resolved into two steps; first that expressed above, second the segregation process. Post collision, post–segregation (re–coloured), quantities will be indicated by the use of a double dagger superscript.

Before proceeding, it is necessary to remark on certain properties of the Guo LBGK model [48]. As mentioned previously, the effect of a fluid body force is partly carried in a re-defined velocity which has a notable consequence for the first moment of the $f_i^\dagger$ as defined in equation (226):

$$\sum_i f^\dagger c_{i\alpha} = \rho u_\alpha^* + f F_\alpha, \qquad f = \frac{1}{2}, \tag{227}$$

which result may be established by taking first moments of the expansion of f_i expressed in (45) and summing on i as follows. Appealing to equation (5) of Guo *et al* [48] and my equations (51) it is possible to identify the first moment of the $f_i^{(1)}$ component in terms of the body force, F. Replacing f_i in the LBGK evolution equation (226) with (45) and taking it's first moments and using Guo's equations (7), (8c), (11), and (17) and observations of Guo *et al* [48]. Equation (227) follows with some straightforward algebra. Note, Guo's time step parameter set to $\Delta t = 1$.

For the usual LBGK model used in Chapter 7, the parameter f introduced in equation (227) would, of course, take the value $f = 1$.

We proceed to develop the macroscopic dynamics arising from the formulaic segregation encapsulated in equations (231) and (232) by attempting to obtain an expression for $\frac{d}{dt}\rho^N$ later in section 8.8.

8.6 Acceleration of Colour Flux

As commented upon earlier in section 4.2.2, the post-propagation, pre-collision, nodal red and blue masses and single colour fluxes in terms of the link based quantities $R_i(\mathbf{r}, t)$ and $B_i(\mathbf{r}, t)$ are defined as:

$$R(\mathbf{r}, t) = \sum_i R_i(\mathbf{r}, t), \qquad B(\mathbf{r}, t) = \sum_i B_i(\mathbf{r}, t), \tag{228}$$

$$q_{R\alpha}(\mathbf{r}, t) = \sum_i R_i(\mathbf{r}, t) c_{i\alpha}, \qquad q_{B\alpha}(\mathbf{r}, t) = \sum_i B_i(\mathbf{r}, t) c_{i\alpha}. \tag{229}$$

The overall colour flux is also defined as:

$$\mathbf{q}(\mathbf{r},t) \equiv \mathbf{q}_R(\mathbf{r},t) - \mathbf{q}_B(\mathbf{r},t), \tag{230}$$

which is further discussed in much greater detail when considering the dynamics of the phase field and the colour flux vector in section 8.8.

It is now possible to define a slightly modified colour segregation after Latva-Kokko and Rothman's form of d'Ortona's segregation [6], [57]. We assume that close to rest the equilibrium distribution $f^{(0)}(\rho,\underline{0})$ can be approximated as ρt_p and take:

$$R_i^{\dagger\dagger} = \frac{R}{R+B} f_i^{\dagger} + \beta \frac{RB}{R+B} t_p cos(\theta_f - \theta_i) \left|\mathbf{c_i}\right|, \tag{231}$$

$$B_i^{\dagger\dagger} = \frac{B}{R+B} f_i^{\dagger} - \beta \frac{RB}{R+B} t_p cos(\theta_f - \theta_i) \left|\mathbf{c_i}\right|, \tag{232}$$

in which θ_f (θ_i) is the angle of the colour field, $\mathbf{f}'(\mathbf{r})$, (link) and $R_i^{\dagger\dagger}$ denotes the post–collision, post segregation value of the momentum density of the red fluid associated with link i. The inclusion, in equation (231) and (232), of the factor $|\mathbf{c_i}|$ makes the resulting algorithm more amenable to analysis whilst preserving the essential ideas of Latva-Kokko and Rothman and d'Ortona *et al* [6], [57]. Whilst the equation (231) essentially accords with equation 9 of Latva-Kokko and Rothman [6] (the latter being an improvement over the original method expressed in equation 8 of d'Ortona *et al* [57]), it will be necessary to revisit certain of Latva-Kokko and Rothmans's results later in this section and also in 8.7, where segregation parameter, β, is considered in more detail.

It is necessary to note two properties of equation (231). First:

$$\sum_i R_i^{\dagger\dagger} = \frac{R}{R+B} \sum_i f_i^{\dagger} = R, \tag{233}$$

where the identity $cos(\theta_f - \theta_i)|\mathbf{c}_i| = cos(\theta_f)c_{ix} + sin(\theta_f)c_{iy}$ has been used and

recalling that $\sum_i t_p c_{i\alpha} = 0$ where $\alpha = x, y$ and that the density is given by the sum of the red and blue fluids. Secondly:

$$\sum_i R_i^{\dagger\dagger} \mathbf{c}_i \equiv \mathbf{q}_R^{\dagger\dagger} = R\mathbf{u}^* + \beta k_2 \frac{RB}{R+B} \widehat{\mathbf{f}'} + f \frac{R}{R+B} \mathbf{F}, \tag{234}$$

where we have used equation (227) and those properties of even lattice moments given in equations (224) and (225), the unit vector $\widehat{\mathbf{f}'}$ is defined in equation (217), $\mathbf{F}$ is any continuum body force and, recall:

$$f = \frac{1}{2}, \quad 1, \tag{235}$$

for Guo's LBGK model variant [48] and standard LBGK respectively. For the D2Q9 lattice, $k_2 = 1/3$ (and $k_4 = 1/9$). For the segregation of the blue component, let $\theta_f \rightarrow (\theta_f + \pi)$ thus making the corresponding blue species equation:

$$\sum_i B_i^{\dagger\dagger} \mathbf{c}_i \equiv \mathbf{q}_B^{\dagger\dagger} = B\mathbf{u}^* - \beta k_2 \frac{RB}{R+B} \widehat{\mathbf{f}'} - f \frac{B}{R+B} \mathbf{F}, \tag{236}$$

Hence, from equation (234) and (236), the net flux of colour vector:

$$\mathbf{q}^{\dagger\dagger} \equiv \mathbf{q}_R^{\dagger\dagger} - \mathbf{q}_B^{\dagger\dagger} = \sum_i \left(R_i^{\dagger\dagger} \mathbf{c}_i - B_i^{\dagger\dagger} \mathbf{c}_i \right), \tag{237}$$

may be written:

$$\mathbf{q}^{\dagger\dagger} = (R-B)\mathbf{u}^* + 2\beta k_2 \frac{RB}{R+B} (cos(\theta_f), sin(\theta_f)) + f\rho^N \mathbf{F}, \tag{238}$$

and replacing quantities R and B, using the definitions expressed in (222) and (223), equation (238) takes the form:

$$\mathbf{q}^{\dagger\dagger} = \rho\rho^N \mathbf{u}^* + \frac{1}{2}\beta k_2 \rho \left(1 - \rho^{N2}\right) \widehat{\mathbf{f}'} + f\rho^N \mathbf{F}. \tag{239}$$

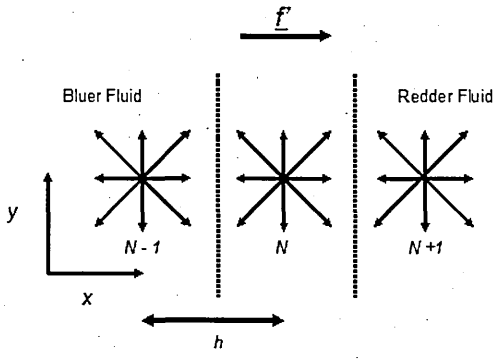

Figure 34: Analysis of a flat, equilibrium interface orientated perpendicular to short lattice links. The lattice links on each of the nodes depicted are indexed as in figure 33. The colour gradient points from left to right; the vertical broken lines represent contours of constant phase–field value.

Now, $\mathbf{q}^{\dagger\dagger} = \rho\rho^N\mathbf{u}^*$ represents a post–collision colour concentration advecting in flow. Neglecting for the moment correlations between $\mathbf{q}$ and $\mathbf{q}^{\dagger\dagger}$, the segregation expressed in equation (231) tends to *accelerate* the colour flux in the direction of the colour field at the rate $\frac{1}{2}\beta k_2\rho\left(1-\rho^{N2}\right)$. We return to this point later in this chapter after equation (293).

8.6.1 Planar Interface at Rest

Before proceeding with our derivation of the interface dynamical equation, we pause to investigate the structure of a flat boundary between rest fluids, formed by rules (231) and (232).

Consider a lattice fluid a rest with uniform density ρ, in the steady state, containing a flat interface parallel to short lattice links in a D2Q9 fluid. This situation is depicted in figure 34, in which two contours of constant phase field parameter ρ^N are represented by broken lines and three adjacent lattice sites are shown. The indexing of the link label, i, for all three sites is as defined in figure 33. Note that an explicit lattice spacing, h, parameter has been introduced for present purposes. The

colour gradient field unit vector $\widehat{f'} = \widehat{x}$. For the situation depicted in figure 34 there is no variation of ρ^N with the y coordinate; furthermore, on grounds of symmetry alone:

$$R_3(x_n, y) = R_5(x_n, y), \quad R_1(x_n, y) = R_7(x_n, y), \quad R_2(x_n, y) = R_6(x_n, y), \qquad (240)$$

where R may be a pre or post collision value and $n = (N-1), N, (N+1)$.

At equilibrium, there is no change of colour content at any site and, henceforth suppressing the y coordinate, we can write the following equilibrium condition for the red density on site N:

$$\begin{aligned} R(x_N) &= 2R_3^{\dagger\dagger}(x_{N-1}) + R_4^{\dagger\dagger}(x_{N-1}) + 2R_2^{\dagger\dagger}(x_N) \\ &+2R_1^{\dagger\dagger}(x_{N+1}) + R_8^{\dagger\dagger}(x_{N+1}) + R_0^{\dagger\dagger}(x_N). \end{aligned} \qquad (241)$$

Using equation (251) for $R_i^{\dagger\dagger}$ with the equilibrium solution $f_i^{\dagger} = t_p\rho$ with appropriate values of t_p and colour field $\widehat{f}$, the last equation may, after some simple algebra, be written:

$$\begin{aligned} 2R(x_N) &= R(x_{N-1})\left(1 + \beta'(\rho - R(x_{N-1}))\right) \\ &+R(x_{N+1})\left(1 - \beta'(\rho - R(x_{N+1}))\right), \end{aligned} \qquad (242)$$

where $\beta' = \beta/\rho$ and the density ρ is considered constant giving $B = (\rho - R)$. Taylor expanding the terms in this equation about x_N, to second order in lattice parameter, h, to give the following definitions:

$$R(x_{N+1}) = R(x_N) + hR'(x_N) + \frac{h^2}{2!}R''(x_N) + ... \qquad (243)$$

Through substitution of (243) into (242) and retaining terms to second order in lattice parameter, h, it is straightforward to obtain the following equation for the

red density, $R(x)$, at equilibrium:

$$\frac{d^2R}{dx^2} - 2\frac{\beta}{h}\frac{dR}{dx} + 2\frac{\beta'}{h}\frac{dR^2}{dx} = 0, \tag{244}$$

which may be integrated once, straightforwardly to obtain a separable first–order differential equation:

$$\frac{dR}{dx} = \frac{2\beta}{h}R\left(1 - \frac{R}{\rho}\right), \tag{245}$$

which, note, differs only superficially from equation (15) of Latva-Kokko and Rothman [6]. The last equation may be readily integrated. On supposing that the site at x_N lies at the centre of an interface, the constants of integration are easily obtained:

$$R(x) = tanh\left(\frac{\beta}{h}(x - x_N)\right). \tag{246}$$

The above may be checked by substitution into equation (244).

On setting $h = 1$, segregation parameter β is seen to be the parameter of the equilibrium interface width. Clearly, the preceding analysis generalizes directly to case of a 3D lattice.

For a D2Q9 lattice, a conceptually identical but somewhat more complicated analysis of the case of an interface parallel to the long links of the lattice shows that the equilibrium state of the interfacial phase field is described by differential equation (244) solution (246).

8.6.2 Curved Interface

We saw in the last section that by considering the flat interface, centred on $x = x_0$, embedded in a rest fluid, a solution for the steady–state spatial variation in the phase field can be obtained (by analytical methods) which may be expressed in

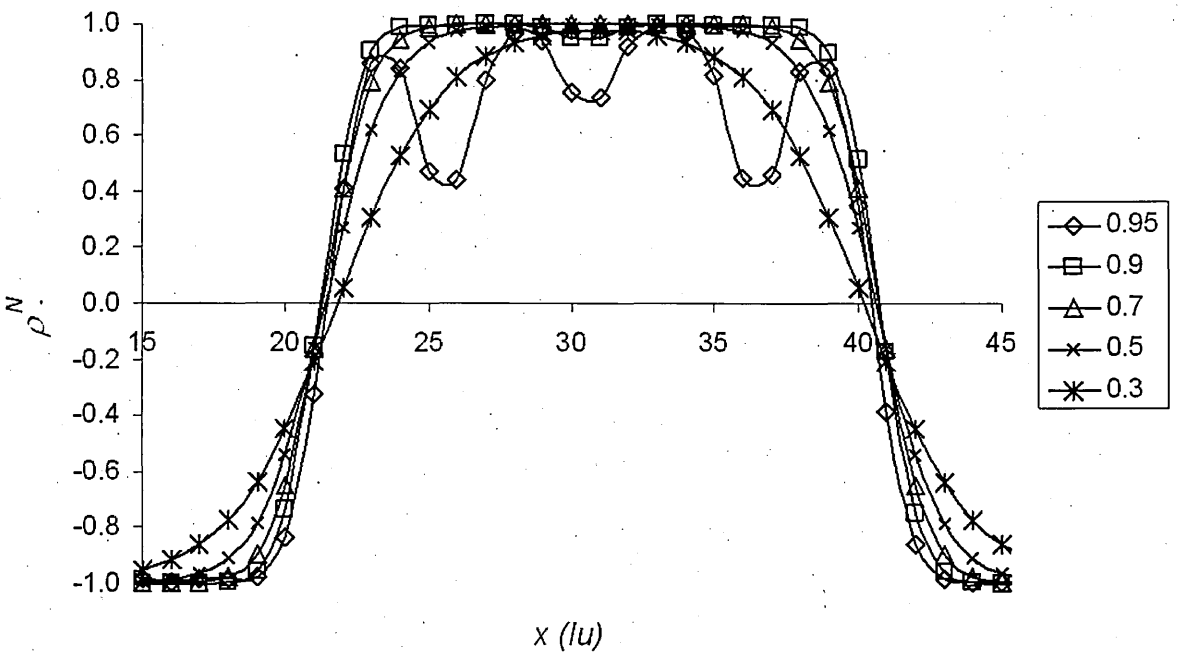

Figure 35: Variation of the phase field parameter, ρ^N, with distance, x, across the equator of a red ($\rho^N = 1$) drop of approximate initial radius 10 lattice units, obtained for a range of segregation parameters β (equation (231) and figure key).

generalised notation as:

$$\rho^N = tanh\left(k(x - x_0)\right), \quad k = \beta. \tag{247}$$

As noted above, this equation applies to both (i) the interface parallel to the short links of a D2Q9 lattice and (ii) the interface parallel to long links of a D2Q9 lattice. Furthermore, for a distributed, circular drop interface, with $0 < \beta < 0.7$, simulation results like those in figure 35 (which correspond $\beta = 0.7$, note) show that the steady interface profile generated using equation (231) is well–approximated (see below) by a phase field parameter variation:

$$\rho^N(s) = tanh(ks), \qquad k = \beta, \tag{248}$$

in which s measures distance in the direction of $\widehat{\mathbf{f}'}$ with $s = 0$ corresponding to the centre of the interface. Supported by these observations, we take in general:

$$\nabla\rho^N \approx k sech^2(ks)\widehat{\mathbf{f}'} = k\left(1-\rho^{N2}\right)\widehat{\mathbf{f}'}, \qquad k = \beta. \tag{249}$$

The parameter k shall be retained as the interface parameter throughout this analysis, setting $k = \beta$ only in final results.

It is important to reiterate the fact that, for the case of a D2Q9 lattice, solution (247) may be shown to describe interfacial orientations with the interface parallel to short lattice links as carried out above and also with the interface parallel to the long lattice links; this demonstrates the isotropy implicit in the distributed interface and accords well with Latva–Kokko and Rothman's observations.

Now, returning to the colour flux, after the segregation between colour components may be written:

$$\mathbf{q}^{\dagger\dagger} = \rho\rho^N\mathbf{u}^* + \frac{1}{2}\frac{\beta}{k}k_2\rho\nabla\rho^N + f\rho^N\mathbf{F}. \tag{250}$$

Where the first term in the right hand side of the above expression relates to the colour concentration advecting with flow and the latter term on the right hand side of equation (249) relates to the acceleration of the colour flux. The result expressed in equations (239) and (250) generalizes straightforwardly into three dimensions:

$$R_i^{\dagger\dagger} = \frac{R}{R+B}f_i^{\dagger} + \beta\frac{RB}{R+B}t_p\widehat{\mathbf{f}'}.\mathbf{c}_i. \tag{251}$$

8.7 Stability of the Interface

For application to the continuum hydrodynamic regime, it is necessary to impose limitations on the range of parameter β in equation (231). Figure 35 shows the variation of the phase field, ρ^N, across the equator of a stationary drop, initial

radius 20 lattice units, for a range of *beta* values. Clearly, interfacial thickness decreases as β increases. However, for $\beta \gtrsim 0.7$, there is no stable steady state. This is manifest in the data of figure 35 in *overjump* behaviour which, for application to the continuum regime with arrested evaporation, represents an instability.

An approximate maximum stable value for segregation parameter, β_{max}, may be obtained from the condition $R_i^{\dagger\dagger} \leqslant f_i^{\dagger}$ and equation (251). We require:

$$\frac{R}{R+B} f_i^{\dagger} + \beta \frac{RB}{R+B} t_p \widehat{\mathbf{f}'}.\mathbf{c}_i \leqslant f_i^{\dagger} \qquad foralli, \tag{252}$$

from which it is straightforward to obtain the inequality:

$$t_p \beta R \widehat{\mathbf{f}'}.\mathbf{c}_i \leqslant f_i^{\dagger}. \tag{253}$$

The upper bound for β must correspond to a maximum of the scalar product $\widehat{\mathbf{f}'}.\mathbf{c}_i$; that is, when $\widehat{\mathbf{f}'}$ is parallel to $\mathbf{c}_i$ for some particular link $i = I$, which is supposed to have the longest possible length l_{max}:

$$t_p \beta_{max} R l_{max} \leqslant f_I^{\dagger}. \tag{254}$$

Approximating $f_I^{\dagger} \approx f_I^{(0)}(\rho, \mathbf{0}) = t_p \rho$ and taking the maximum possible value of R (which is close to ρ) provides the required maximum stable value for β, consistent with arrested evaporation. For a D2Q9 lattice, $l_{max} = \sqrt{2}$, so, for our particular simulations:

$$\beta_{max} \lesssim \frac{1}{\sqrt{2}} = 0.71, \tag{255}$$

which value agrees very well indeed with simulations at low Re, Ca. It is important to note that the practical upper bound approximated in this way may be reduced in other regimes of more rapid multi–component flow, when the f_is depart further from their equilibrium values.

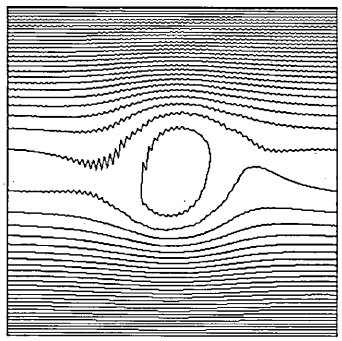

Figure 36: Contours of constant value for the rectangular stream function obtained for a two-dimensional drop with numerical segregation, exposed to a small, symmetric shear flow.

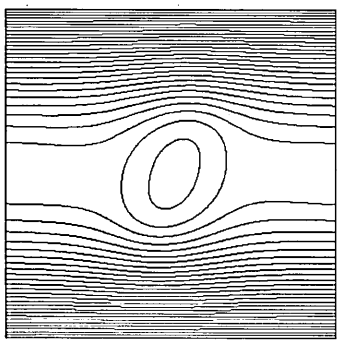

Figure 37: Contours of constant value for the rectangular stream function obtained for a two-dimensional drop with the segregation method after d'Ortona [57], analysed in section 8.5. Again the drop was exposed to a small, symmetric shear flow, identical to that used for the data of figure 36.

Compared with the formulaic segregation expressed in equation (251), numerical segregation [24] produces much narrower interfaces; typically its phase field switches in ~ 1.5 lattice spacings see Chapter 7. An obvious question arises around replacing a numerical segregation by the formulaic segregation; whilst numerical segregation is affected by pinning, and computationally it is much slower, surely the sharper interface of numerical segregation better accords with the continuum concept of a discontinuous interface? In fact, a close examination of the rectangular stream function of symmetrically sheared drops, maintained by numerical segregation (figure 36) and formulaic segregation (figure 37) provides a reply.

Figures 36 and 37 relate otherwise identical, slightly sheared drops maintained (respectively) by numerical segregation and formulaic segregation. The last open and first closed contours have been selected closely to *sandwich* the interface. In figure 36, this pair of contours exhibit oscillations of an amplitude which, when resolved in the direction of the local interface normal, extend over ≤ 5 lattice spacings exposing a noisy velocity field in the region of the interface. Such noise is absent from the equivalent results of figure 37. In fact, it is possible to examine the precise hydrodynamic boundary conditions recovered in this interfacial region and to develop, for formulaic segregation, simple, robust, highly adaptable and very effective algorithmic corrections which effectively sharpen the interface; this algorithmic modification is to be covered in detail in Chapter 9. Furthermore, even though the phase-field interface is sharper, the hydrodynamic interface obtained with numerical segregation (as defined by properties of the velocity field) is diffuse over a distance comparable to the phase field variation obtained with formulaic segregation. So, with numerical segregation, the interface is not consistently sharper. It is appropriate also to emphasize that, this problem only becomes more acute as drop surface tension (deformation) increases (decreases).

8.8 Dynamics of the Phase Field

Having gathered several key results we find ourselves in a position to pursue the key objective of this chapter, to derive an expression for the dynamics of the phase field.

The dynamics of numerical species segregation has not, to my knowledge, been derived. It is a considerable advantage that d'Ortona's formulaic method allows one to derive the dynamics of the phase field, which result form use of its characterizing equation (231). It is therefore appropriate to consider the equation of motion of the phase field, ρ^N.

Unlike numerical segregation (or re–colouring) strategies [24], the formulaic seg-

regation method of d'Ortona *et al* [57] is amenable to analysis. To facilitate this analysis, formulaic segregation is described formally using a single relaxation time LBGK evolution equation with collision parameter ω_c and the corresponding macroscopic dynamics are obtained by a Chapman-Enskog expansion process. The segregation encapsulated in equation (231) represents a special case of the following, generalized formulaic segregation or 'colour collision':

$$R_i(\mathbf{r}+\mathbf{c}_i, t+1) = R_i(\mathbf{r}, t) - \omega_c \left(R_i(\mathbf{r}, t) - R_i^{(0)} \right), \tag{256}$$

with;

$$R_i^{(0)} \equiv \frac{R}{R+B} f_i^{\dagger}(\mathbf{r}, t) + \beta \left(\frac{1}{R} + \frac{1}{B} \right)^{-1} t_p cos(\theta_f(\mathbf{r}, t) - \theta_i) \left| \mathbf{c}_i \right|, \tag{257}$$

in which $R = R(\mathbf{r}, t)$. It is important to emphasize that equation (257) is a generalized segregation; setting $\omega_c = 1$ will enable us to extract the macroscopic dynamics of the particular formulaic segregation after references [57] and [6]; the aim here is not to consider a generalized segregation process. Note that the above equation (257) is the same as that expressed in (231) with the bracket term re-expressed as a reciprocal and that $f_i^{\dagger}$ includes any source term contribution; see equation (226).

It is assumed that it is possible to expand the $R_i(\mathbf{r}, t)$ about equilibrium:

$$R_i(\mathbf{r}, t) = \sum_n \epsilon^n R_i^{(n)}, \tag{258}$$

where ϵ is a Chapman-Enskog parameter equivalent to δ used previously. It is now straightforward to obtain the following three results for the moments of $R_i^{(0)}$; first:

$$\sum_i R_i^{(0)} = R(\mathbf{r}, t) \equiv \sum_i R_i(\mathbf{r}, t), \tag{259}$$

$$\sum_i R_i^{(0)} \mathbf{c}_i = R\mathbf{u}^* + k_2 \beta \left(\frac{1}{R} + \frac{1}{B} \right)^{-1} \widehat{\mathbf{f}'} + f \frac{R}{R+B} \mathbf{F} \equiv \mathbf{q}_R^{\dagger\dagger}, \tag{260}$$

$$\sum_i R_i^{(0)} c_{i\alpha} c_{i\beta} = \frac{R}{R+B} \sum_i f_i^{\dagger} c_{i\alpha} c_{i\beta} \equiv \Lambda_{R\alpha\beta}, \tag{261}$$

where equation (234) has been used and, in equation (261), the fact that odd moments of the lattice distribution function are zero has been utilised. Defining:

$$\Lambda_{\alpha\beta} \equiv \Lambda_{R\alpha\beta} - \Lambda_{B\alpha\beta}. \tag{262}$$

Equations (260)..(262) shall be used shortly. For the moment, note that equation (259) implies that:

$$\sum_i R_i^{(n>1)} = 0, \tag{263}$$

but, however, it is important to note that it is not possible to infer a corresponding result for the first moment of the $R_i^{(0)}$: $\sum_i R_i^{(n>1)} c_{i\alpha} \neq 0$. From equations (256)...(263), only equation (260) changes under the exchange $B \leftrightarrow R$:

$$\sum_i B_i^{(0)} \mathbf{c}_i = B\mathbf{u}^* - k_2\beta \left(\frac{1}{R} + \frac{1}{B}\right)^{-1} \widehat{\mathbf{f}'} + f\frac{B}{R+B}\mathbf{F} \equiv \mathbf{q}_B^{\dagger\dagger}, \tag{264}$$

and so:

$$\begin{aligned} \sum_i \left(R_i^{(0)} - B_i^{(0)}\right) \mathbf{c}_i &= \rho\rho^N \mathbf{u}^* + 2k_2\beta \left(\frac{1}{R} + \frac{1}{B}\right)^{-1} \widehat{\mathbf{f}'} + f\rho^N \mathbf{F} \\ &= \mathbf{q}^{\dagger\dagger}, \end{aligned} \tag{265}$$

where, in the second part of the equality, equation (238) has been used.

Proceeding in the fashion of Chapman-Enskog as outlined in section 4.1.2; that is, by using a Taylor expansion of the evolution equation (256) about $\mathbf{r}, t$ and a subsequent scale expansion of, the time derivative [3]:

$$\partial_t = \partial_{t0} + \epsilon\partial_{t1}, \qquad \partial_x = \epsilon\partial_x, \tag{266}$$

in addition to section 4.1.2 discussion is contained in [3] and [27]. Note that in

the form of Chapman Enskog expansion used here (which is entirely equivalent to that used in section 4.1.2 the space derivative is also given a scale label, ϵ). The Chapman-Enskog expansion then generates the following equations; at $o(\epsilon)$:

$$\partial_{t0} R_i^{(0)} + c_{i\alpha}\partial_\alpha R_i^{(0)} = -\omega_c R_i^{(1)}, \tag{267}$$

and at $o(\epsilon^2)$:

$$\partial_{t1} R_i^{(0)} + \left(1 - \frac{\omega_c}{2}\right)\left(\partial_{t_0} + c_{i\alpha}\partial_\alpha\right) R_1^{(1)} = -\omega_c R_i^{(2)}. \tag{268}$$

Now, consider the $o(\epsilon)$ dynamics. Summing equation (267) over i, using equation (263) and definition (260), we obtain:

$$\partial_{t0} R + \nabla.\left(R\mathbf{u}^* + k_2\beta\left[\frac{1}{R} + \frac{1}{B}\right]^{-1}\widehat{\mathbf{f}'} + f\frac{R}{R+B}\mathbf{F}\right) = 0, \tag{269}$$

and similarly, for the B component:

$$\partial_{t0} B + \nabla.\left(B\mathbf{u}^* - k_2\beta\left[\frac{1}{R} + \frac{1}{B}\right]^{-1}\widehat{\mathbf{f}'} + f\frac{B}{R+B}\mathbf{F}\right) = 0. \tag{270}$$

Subtracting equations (269) and (270), there results the following macroscopic dynamics for the phase field scalar at $o(\epsilon)$:

$$\partial_{t0}\rho\rho^N + \partial_\alpha.\rho\rho^N u_\alpha^* = -2k_2\beta\partial_\alpha\left(\left[\frac{1}{R} + \frac{1}{B}\right]^{-1}\widehat{f}'_\alpha\right) - f\partial_\alpha\left(\rho^N F_\alpha\right). \tag{271}$$

Multiply equation (267) by link velocity component $c_{i\beta}$, sum the resulting equation over i and follow a similar analysis using equation (261) as opposed to (260), to obtain (still at $o(\epsilon)$):

$$\begin{aligned}\partial_{t0}\left(\rho\rho^N u_\beta^* + 2k_2\beta\left(\frac{1}{R} + \frac{1}{B}\right)^{-1}\widehat{f}'_\beta + f\rho^N F_\beta\right)\\ + \partial_\alpha\Lambda_{\alpha\beta} = -\omega_c\sum_i\left(R_i^{(1)} - B_i^{(1)}\right)c_{i\beta},\end{aligned} \tag{272}$$

in the right hand side of equation (272) make the approximation $C_i^{(1)} \approx (C_i - C_i^{(0)})$ $(C = R, B)$ and use equations (230) and (265); equation (272) now becomes:

$$\partial_{t0}\left(\rho\rho^N \mathbf{u}^* + 2k_2\beta\left(\frac{1}{R}+\frac{1}{B}\right)^{-1}\widehat{\mathbf{f}}' + f\rho^N\mathbf{F}\right) + \partial_\beta\Lambda_{\alpha\beta} = \omega_c(\mathbf{q}^{\dagger\dagger} - \mathbf{q}). \tag{273}$$

Next consider the $o(\epsilon^2)$ dynamics. In equation (268), substitute for $R_i^{(1)}$ from equation (267) and again sum on i; after some algebra, we obtain:

$$\partial_{t1}R + \left(\frac{1}{2}-\frac{1}{\omega_c}\right)\left(\partial_{t0}^2 R + 2\partial_{t0}\partial_\alpha\left(\sum_i R_i^{(0)}c_{i\alpha}\right) + \partial_\alpha\partial_\beta\Lambda_{R\alpha\beta}\right) = 0, \tag{274}$$

with a corresponding equation for B, which, when subtracted from equation (274), yields:

$$\partial_{t1}\rho\rho^N + \left(\frac{1}{2}-\frac{1}{\omega_c}\right)\left(\partial_{t0}^2\rho\rho^N + 2\partial_{t0}\partial_\alpha\left(\rho\rho^N u_\alpha^* + 2k_2\beta\left(\frac{1}{R}+\frac{1}{B}\right)^{-1}\widehat{f}'_\alpha + f\rho^N F_\alpha\right) + \partial_\alpha\partial_\beta\Lambda_{\alpha\beta}\right) = 0, \tag{275}$$

where equation (265) has been used. By using equations (273) and (271) it is possible, again after some algebra, to obtain from the last equation the following dynamics for the phase field at $o(\epsilon^2)$:

$$\partial_{t1}\rho\rho^N = \left(1-\frac{\omega_c}{2}\right)\nabla.\left(\mathbf{q}^{\dagger\dagger} - \mathbf{q}\right). \tag{276}$$

Turning attention now to the derivation of an expression for colour flux vector, $\underline{q}$ to use in the right hand side of the above.

Substitute equations (229) into definition (230) and reverse the propagate step

to obtain:

$$\mathbf{q}(\mathbf{r},t)=\sum_i\left(R_i^{\dagger\dagger}\left(\mathbf{r}-\mathbf{c}_i,t-1\right)-B_i^{\dagger\dagger}\left(\mathbf{r}-\mathbf{c}_i,t-1\right)\right)\mathbf{c}_i. \tag{277}$$

Substituting with equation (251) and using definitions (222) and (223) it is possible to obtain straightforwardly:

$$\begin{aligned}q_\alpha(\mathbf{r},t)=&\sum_i\rho^N\left(\mathbf{r}-\mathbf{c}_i,t-1\right)f_i^\dagger\left(\mathbf{r}-\mathbf{c}_i,t-1\right)c_{i\alpha}+\\&\frac{\beta}{2}\sum_i t_p\rho\left(\mathbf{r}-\mathbf{c}_i,t-1\right)\left(1-\rho^{N2}\left(\mathbf{r}-\mathbf{c}_i,t-1\right)\right)\widehat{f}'_\beta\left(\mathbf{r}-\mathbf{c}_i,t-1\right)c_{i\alpha}c_{i\beta},\end{aligned} \tag{278}$$

in which, note, $f_i^\dagger$ contains the body–force source term contribution. Equation (278) shall be treated term by term in the following analysis.

In continuum applications, the interfacial phase field is always close to equilibrium; accordingly its time variation shall be neglected. Moreover, bearing in mind the form of the gradient already present in the right hand side of equation (276), it is necessary to work only to first order in space derivatives to obtain second order accuracy in the Chapman-Enskog expansion. Accordingly, the first term in the right hand side of equation (278) is treated by Taylor-expanding ρ^N and $f_i^\dagger$ about $\mathbf{r}$, to obtain the following expression:

$$\begin{aligned}\sum_i\rho^N(r-c_i,t-1)f_i^\dagger(r-c_i,t-1)c_{i\alpha}=&\sum_i(\rho^N-c_{i\beta}\delta_\beta\rho^N+\frac{1}{2!}c_{i\gamma}c_{i\beta}\delta_\gamma\delta_\beta\rho^N)\\&(f_i^\dagger-c_{i\delta}\delta_\delta f_i^\dagger+\frac{1}{2!}c_{i\delta}c_{i\phi}\delta_\delta\delta_\phi f_i^\dagger)c_{i\alpha}.\end{aligned} \tag{279}$$

Multiplying out the above and neglecting all second derivatives leads to the following

expression:

$$\rho^N\left(\sum_i f_i^\dagger c_{i\beta}\right) - \rho^N\partial_\alpha\left(\sum_i f_i^\dagger c_{i\alpha}c_{i\beta}\right) - \left(\partial_\alpha\rho^N\right)\sum_i f_i^\dagger c_{i\alpha}c_{i\beta} + o\left(\delta^3\right). \quad (280)$$

In section 8.5, equation (227), it was stated that $\sum_i f^\dagger c_{i\alpha} = \rho u_\alpha^* + fF_\alpha$, with $f = 1, \frac{1}{2}$ for the standard and Guo variants respectively. Expressions in macroscopic quantities for the second and third terms in expression (280) may be obtained in like manner, for the Guo variant: substitute for $f_i^\dagger$ using definition (226), replace f_i with $\left(f_i^{(0)} + \epsilon f_i^{(1)} + ..\right)$ take second moments of the resulting equation and sum on i. Then substitute using equations (7) of Guo *et al* [48] (Guo's parameter A is set to zero for mass conservation, note). Guo proceeds to make the following identities:

$$\begin{aligned}\Pi_{\alpha\beta}^{(0)} &= \sum_i c_{i\alpha}c_{i\beta}f_i^{(0)},\\ \Pi_{\alpha\beta}^{(1)} &= \sum_i c_{i\alpha}c_{i\beta}f_i^{(1)},\end{aligned} \quad (281)$$

and to provide expressions for them in equations 12 and 16 of guo *et al* [48] with these and with Guo's time-step parameter $\Delta t = 1$ and appealing to equation 16 and equation 12 of Guo *et al* [48], it is possible, after some algebra to obtain:

$$\begin{aligned}\sum_i f_i^\dagger c_{i\alpha}c_{i\beta} =& c_s^2\rho\delta_{\alpha\beta} + \rho u_\alpha^* u_\beta^* + \\ & \frac{1}{2}(\omega-1)\left(u_\alpha^* F_\beta + u_\beta^* F_\alpha\right) + 2\left(1-\frac{1}{\omega}\right)c_s^2\rho S_{\alpha\beta}^*.\end{aligned} \quad (282)$$

For the standard model the equivalent result is easily obtained as follows from equations (276) and (84):

$$\sum_i f_i^\dagger c_{i\alpha}c_{i\beta} = \sum_i f_i^{(0)}c_{i\alpha}c_{i\beta} + \sum_i f_i^{(1)\dagger}c_{i\alpha}c_{i\beta} + \sum_i \phi_i c_{i\alpha}c_{i\beta}. \quad (283)$$

Note, $\phi_i = k_\alpha c_{i\alpha}\partial_\alpha\rho$. Using equation (41) and the fact that the odd lattice moments

in the last term vanishes:

$$\sum_i f_i^\dagger c_{i\alpha} c_{i\beta} = \rho c_s^2 \delta_{\alpha\beta} + \rho u_\alpha u_\beta + \sum_i f_i^{(1)\dagger} c_{i\alpha} c_{i\beta}, \tag{284}$$

appeal to equation (88) and equation (84) in Chapter 4 then gives, for the standard model:

$$\sum_i f_i^\dagger c_{i\alpha} c_{i\beta} = c_s^2 \rho \delta_{\alpha\beta} + \rho u_\alpha u_\beta + 2(1-\tau) c_s^2 \rho S_{\alpha\beta}. \tag{285}$$

The second term in the right hand side of equation (278) is first transformed using (i) (249) to replace the factor $\left(1-\rho^N\left(\mathbf{r}-\mathbf{c}_i, t-1\right)^2\right)\widehat{f}'$ with $|\nabla\rho^N|/k$ evaluated at position $\mathbf{r}-\mathbf{c}_i$ and (ii) the definition of $\widehat{f}'_\beta$, in equation (217). After Taylor expansion it is possible to express the right hand side of equation (278) as:

$$\begin{aligned} &\frac{\beta}{2}\sum_i t_p \rho\left(\mathbf{r}-\mathbf{c}_i, t-1\right)\left(1-\rho^N\left(\mathbf{r}-\mathbf{c}_i, t-1\right)^2\right)\widehat{f}'_\beta\left(\mathbf{r}-\mathbf{c}_i, t-1\right) c_{i\alpha} c_{i\beta} \\ &= \frac{\beta}{2}\sum_i \frac{1}{k} t_p (\rho + c_{i\alpha}\delta_\alpha \rho + \ldots)(\delta_\beta \rho^N + c_{i\alpha}\delta_\alpha(\delta_\beta \rho^N)) c_{i\alpha} c_{i\beta}. \end{aligned} \tag{286}$$

Recalling that terms cubic in c_i are zero and appealing to equations (224) yields:

$$\frac{1}{2}\frac{\beta}{k} k_2 \rho \partial_\alpha \rho^N + o\left(\delta^3\right), \tag{287}$$

where, recall $k = \beta$.

Using expressions (280), (287), equation (278) and (282) it is possible to obtain, correct to first order spatial gradients, the following expression for q_α in the Guo

model where cancellation has been carried out using the chain rule:

$$\begin{aligned} q_\alpha(\mathbf{r},t) =& \rho\rho^N u^*_\alpha - c_s^2\partial_\alpha\left(\rho\rho^N\right) + \frac{1}{2}\rho^N F_\alpha + \frac{1}{2}\frac{\beta}{k}k_2\rho\partial_\alpha\rho^N \\ & - \partial_\beta\left(\rho\rho^N u^*_\alpha u^*_\beta\right) - \frac{1}{2}\partial_\beta\left(\rho^N(\omega-1)\left[u^*_\alpha F_\beta + u^*_\beta F_\alpha\right]\right) \\ & - 2\left(1-\frac{1}{\omega}\right)c_s^2\partial_\beta\left(\rho^N S_{\alpha\beta}\right), \end{aligned} \tag{288}$$

in which we again emphasize the terms in the second line are neglected. Since, when used in the right hand side of equation (276) (as will be carried out in the following steps), they will produce second order spatial gradients of terms quadratic in u_α and F_α. The corresponding result for the standard model is easily established using equations, again from expressions (280), (287), equation (278) and, now, (285):

$$q_\alpha(\mathbf{r},t) = \rho\rho^N u_\alpha - c_s^2\partial_\alpha\left(\rho\rho^N\right) + \rho^N F_\alpha\partial_\alpha\rho^N + \frac{1}{2}\frac{\beta}{k}k_2\rho\partial_\alpha\rho^N - \left(1-\frac{1}{\omega}\right)c_s^2\partial_\beta\left(\rho^N S_{\alpha\beta}\right). \tag{289}$$

When combined with equation (250) it is possible to re-express the right hand side of equation (276) to obtain a simple result:

$$\partial_{t1}\rho\rho^N = \frac{1}{2}c_s^2\nabla^2\left(\rho\rho^N\right), \tag{290}$$

in which equation we have now set $\omega_c = 1$ and used the fact that $k = \beta$, note. To combine the $o(\epsilon)$ and $o(\epsilon^2)$ time scales, add equations (271) and (ϵ times) equation (290), invoke identity (266) and identities (222) and (223) and use the facts that $k = \beta$ and, in LBGK models, $k_2 = c_s^2$ [27] to obtain:

$$\partial_t\rho\rho^N + \nabla.\left(\rho\rho^N\mathbf{u}^*\right) = -f\nabla.\left(\rho^N\mathbf{F}\right) + \frac{1}{2}c_s^2\nabla.\left(\rho^N\nabla\rho\right). \tag{291}$$

Now, equation (291) describes the behaviour of the phase field for multi–component LBGK model based upon Guo's enhanced variant ($f = 1/2$) and for the standard, unmodified model ($f = 1$). The final form of the phase field dynamics for these two

cases are now considered separately.

Consider Guo's LBGK model, characterized by $f = 1/2$. On utilizing the continuity equation 18a of reference [48], for the lattice fluid:

$$\partial_t \rho + \nabla.(\rho \mathbf{u}^*) = 0, \tag{292}$$

it is straightforward using the product rule to manipulate equation (291) into a form containing the material derivative:

$$\frac{D}{Dt}\rho^N = \frac{1}{2\rho}\nabla.\left(c_s^2 \rho^N \nabla\rho - \rho^N \mathbf{F}\right), \tag{293}$$

where, note, interface force, $\mathbf{F}$, (see equation (215) and associated discussion) is impressed on the fluid only in the interfacial region.

For the standard LBGK model, it is possible to show that the presence of a variable fluid body force, $\mathbf{F}$, generates additional terms in the lattice fluid continuity equation of for the lattice fluid [46], [61], which becomes:

$$\partial_t \rho + \nabla.(\rho \mathbf{u}) = -\frac{1}{2}\nabla \mathbf{F}, \tag{294}$$

which now implies phase field dynamics:

$$\frac{D}{Dt}\rho^N = -\frac{1}{2\rho}\nabla.\left(\rho^N \mathbf{F}\right) + \rho^N \frac{1}{2\rho}\nabla.(\mathbf{F}) = -\frac{1}{2\rho}\mathbf{F}.\nabla\rho^N. \tag{295}$$

In a continuum fluid, the interface is subject to a kinematic condition which requires that it (the interface) move at the same speed as the local fluids. For a point in the interface, identified by a chosen value of ρ^N, to advect with local flow requires a condition $\frac{D\rho^N}{Dt} = 0$. It follows that, in either of the models represented by equations (293) and (295) (and, indeed, in all multi–component LB) the interface is accelerated relative to the local fluid, in fact by an amount independent of the

local flow and interfacial tension, determined mainly by local phase field gradients. Qualitatively at least, these observations accord with the observations of Latva-Kokko and Rothman [6]. Approximating $\rho = constant$, the value of ρ^N, and hence the term in the right hand side of each of equations (293) and (295), clearly varies across the interfacial region indicating un-physical, relative movement between the phase field and the lattice fluid in both models.

To restore this lack of a kinematic condition attempts are made to limit the differential motion of the lattice fluid and the phase field on closed contour $\rho^N = 0$; the latter appears to be the lattice region onto which efforts (algorithmic extensions) designed more strongly to promote a kinematic condition are best concentrated. None of the several flavours of the LB interface strategies currently in use explicitly impose a kinematic condition. In the following chapter, Chapter 9, attempts are made to address the issue of simple, portable and effective kinematic condition, independent of the background interface model.

8.9 Local Expression For Colour Gradient

During the course of this work attempts were made to derive a local expression for the colour gradient. In the following section, attempts are made to localize the colour gradient. The results reported here are done so as a matter of course to report on general progress and may be omitted if the reader desires.

Introduced in this section is a local expression for $\mathbf{f}'(\mathbf{r})$ which provides a computationally efficient expression suitable for use with a formulaic segregation rule (expressed in equations (231) and (251)) when the local interfacial curvature is not too great or changing too fast (see the discussion at the end of this section).

For red and blue fluids it is possible to define a local *colour field*:

$$\mathbf{g}'(\mathbf{r},t) \equiv -\sum_i \left(R_i(\mathbf{r},t) - B_i(\mathbf{r},t)\right)\mathbf{c}_i, \tag{296}$$

in which red and blue densities are pre–collision values, note. If propagation is taken to be instantaneous, it follows that $R_i(\mathbf{r},t) = R_i^{\dagger\dagger}(\mathbf{r}-\mathbf{c}_i,t-1)$ (and also that $f_i(\mathbf{r},t) = f_i^{\dagger}(\mathbf{r}-\mathbf{c}_i,t-1)$), accordingly:

$$\mathbf{g}'(\mathbf{r},t) \equiv -\sum_i \left(R_i^{\dagger\dagger}(\mathbf{r}-\mathbf{c}_i,t-1) - B_i^{\dagger\dagger}(\mathbf{r}-\mathbf{c}_i,t-1)\right)\mathbf{c}_i. \tag{297}$$

Substitute from equation (231) into equation (297) and use the identities (222) and (223) to obtain:

$$\mathbf{g}'(\mathbf{r},t)_\alpha = S_\alpha^{(1)} + S_\alpha^{(2)}, \tag{298}$$

where:

$$S_\alpha^{(1)} = -\sum_i \rho^N(\mathbf{r}-\mathbf{c}_i,t-1) f_i(\mathbf{r},t) c_{i\alpha}, \tag{299}$$

as previously, the fact that $f_i^{\dagger}(\mathbf{r}-\mathbf{c}_i,t-1) = f_i(\mathbf{r},t)$ is used and:

$$S_\alpha^{(2)} = -\frac{\beta}{2}\sum_i t_p \rho(\mathbf{r}-\mathbf{c}_i,t-1)\left[1-\left(\rho^{N2}(\mathbf{r}-\mathbf{c}_i,t-1)\right)\right] cos(\theta_g(\mathbf{r}-\mathbf{c}_i,t-1)-\theta_i)c_{i\alpha}, \tag{300}$$

in which $\theta_g(\mathbf{r})$ is the direction of the colour field. Using the approximation in equation (249), equation (300) may be transformed:

$$S_\alpha^{(2)} = -\frac{\beta}{2k}\sum_i t_p \rho(\mathbf{r}-\mathbf{c}_i,t-1)\left(\partial_x \rho^N|_{\mathbf{r}-\mathbf{c}_i,t-1} c_{ix}c_{i\alpha} + \partial_y \rho^N|_{\mathbf{r}-\mathbf{c}_i,t-1} c_{iy}c_{i\alpha}\right). \tag{301}$$

Henceforth neglect explicit time dependence. Noting that both equations (301) and (299) remain true in D3QN. The right hand side of both (equations (301) and (299)) are now expanded about $\mathbf{r}$ and, using the expressions (224) and (225) for second and fourth moments of the lattice vectors, after some algebra, it is possible

to show that:

$$S_\alpha^{(1)} = -\rho^N(\mathbf{r})\rho(\mathbf{r})u_\alpha^* + \partial_\beta \rho^N(\mathbf{r}) \sum_i f_i(\mathbf{r}) c_{i\ \alpha} c_{i\ \beta}, \tag{302}$$

$$S_\alpha^{(2)} = -\frac{\beta}{2k} k_2 \rho(\mathbf{r}) \partial_\alpha \rho^N(\mathbf{r}) - \frac{\beta}{2k} k_4 \left\{ \partial_\alpha \rho(\mathbf{r}) \nabla^2 \rho^N(\mathbf{r}) + 2\partial_\theta \rho(\mathbf{r}) \partial_\alpha \partial_\theta \rho^N(\mathbf{r}) \right\}. \tag{303}$$

The quantity $\sum_i f_i(\mathbf{r}) c_{i\alpha} c_{i\beta}$ in equation (302) may be replaced by the sum of the zeroth and first-order momentum flux (stress) tensors; wishing only to retain terms linear in gradient quantities we make the approximation:

$$\sum_i f_i(\mathbf{r}) c_{i\alpha} c_{i\beta} \approx \sum_i f_i^{(0)}(\mathbf{r}) c_{i\alpha} c_{i\beta} = c_s^2 \rho + \rho u_\alpha^* u_\beta^*, \tag{304}$$

where c_s is the speed of sound. Equations (302) and (303) may now be approximated as:

$$S_\alpha^{(1)} \simeq -\rho^N(\mathbf{r})\rho(\mathbf{r})u_\alpha^* - \rho^N(\mathbf{r})\rho(\mathbf{r}) f F_\alpha + c_s^2 \rho(\mathbf{r}) \partial_\alpha \rho^N(\mathbf{r}), \tag{305}$$

$$S_\alpha^{(2)} = -\frac{\beta}{2k} k_2 \rho(\mathbf{r}) \partial_\alpha \rho^N(\mathbf{r}). \tag{306}$$

Where, since F_α is dependant on high order gradients, the term $\rho^N(\mathbf{r})\rho(\mathbf{r}) f F_\alpha$ is dropped from the analysis. Equation (298) now yields the following local approximation to the colour gradient:

$$\nabla \rho^N(\mathbf{r}) = \frac{1}{A} \left(\mathbf{g}' + \rho(\mathbf{r}) \rho^N(\mathbf{r}) \mathbf{u}^* \right), \qquad A \equiv \rho(\mathbf{r}) \left(c_s^2 - \frac{\beta}{2k} k_2 \right). \tag{307}$$

From simulations in D2Q9, for which $c_s^2 = 1/3$, with segregation parameter $\beta = 0.7$, $\rho(\mathbf{r}) = 2$, Measurements of the parameter $1/A = 2.37$, compared with a calculated value of 3.14. Whilst use of the above expression is found to produce instabilities when the local interfacial curvature, $K > 0.1$, it does underwrite a useful increase in the execution speed, determined, of course, by the total amount of interface in the simulation. For a drop of initial radius 20 lattice units, figure 39

shows a surface plot of the value of $\nabla\rho^N(\mathbf{r})$ measured using equation (307).

8.10 Results

Reduction in the pinning of advecting drops, resulting from the use of this re–colouring strategy has been reported by Latva-Kokko and Rothman [6] elsewhere. Whilst the latter uses a different multi–component LB model (the surface tension generating algorithm of Gunstensen [24] is used), Latva-Koko and Rothman effectively demonstrate that it is the re–colouring or segregation steps which lie at the root of an improvement in drop advection, or Galilean invariance properties. Not surprisingly, very similar improvements were observed with the particular LB model described here, in this section, I shall detail some other improvements associated with macroscopic fluid-interfacial properties of the model, which accrue from the use of a fomulaic segregation procedure.

Use of the segregation of section 8.5 is, alone, responsible for a considerable reduction in the un-physical spurious velocity activity (or interfacial micro-current), created by the interface. Micro-current activity was measured as follows. An equilibrated drop of initial radius 20 lattice units was placed on a square lattice, bounded using the second-order accurate boundary closure method described in Chapter 5 for no-slip conditions. The measured velocity modulus was averaged over annular lattice samples of increasing average radius, centred on the drop centre of mass. This activity average is denoted A. This procedure was performed on otherwise identical drops (surface tension parameter and the interfacial pressure steps) maintained (i) by numerical component segregation [24] and (ii) by the formulaic segregation method of section 8.5. Figure 38 shows the value of A as a function of distance from the drop centre, for (upper line) the numerical and (lower line) the formulaic segregation method. Note the use of a logarithmic ordinal scale. The figure insets show the corresponding steady-state micro-current flow fields, by means of a stream

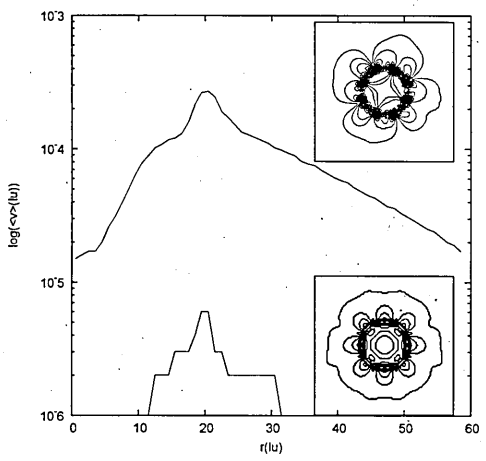

Figure 38: Average interfacial micro-current activity, $\langle v \rangle$, measured in lattice units, plotted as a function of radial distance from the drop centre, r, measured in lattice units, for numerical segregation (upper curve) and for the distributed segregation method (lower curve) for otherwise identical drops. Note the use of the logarithmic scale on the ordinate. There is a difference of almost two orders of magnitude in the micro-current activity.

function plot.

For $\beta < 0.71$, stable interface profiles were found to be smooth and well-approximated by a variation $\rho^N = tanh(ks)$ with s distance in the direction of $\nabla\rho^N$ and $k = \beta$, the segregation parameter introduced in equation (231) (see figure 35). This fact, together with the smooth variation of ρ^N, encourages one to seek a local expression for the gradient in ρ^N, the existence of which would make the algorithm entirely local and very efficient.

As shown previously in section 8.9, it is possible to derive a local expression for $\nabla\rho^N$, which was found to work poorly for highly curved (or small) drops but surprisingly well for drops with curvature $K < 0.1$. Figure 39 shows the key property of phase field gradient for a circular (drop) interface, initial radius 20 lattice units, simulated using the local approximations derived in section 8.9. In figure 39, only lattice sites with both red and blue fluids present have a defined interfacial normal and phase field gradient. Note that the peak value of $\nabla\rho^N$ corresponds to $\rho^N = 0$.

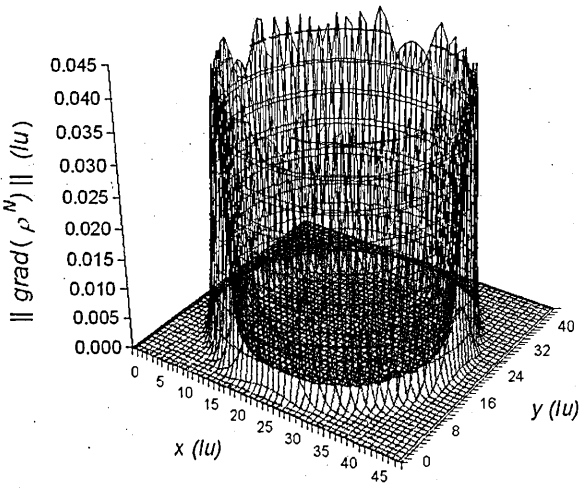

Figure 39: Variation of the phase field gradient for a red drop immersed in a blue fluid (z-axis) versus position. $\nabla\rho^N$ (z-axis) was calculated using the local approximation, given in equation (307), derived in section 8.9. The drop initial radius was 20 lattice units. Only lattice sites with both red and blue fluids present have a defined interfacial phase field gradient. Note that the peak value of $\nabla\rho^N$ corresponds to $\rho^N = 0$.

8.11 Conclusion

The work covered in this section is a presentation of a multi–component lattice Boltzmann simulation applied in the continuum approximation of fluid mechanics, to completely immiscible fluids, paying particular attention to the component segregation part of the underlying algorithm. The principal practical outcome of this section of work is the result that the overall algorithm is found to produce a very low level of micro-current activity indeed; the principal methodological result of this section is the derivation of the dynamics of a component index, or phase field, which have here been obtained for the formulaic segregation method of d'Ortona *et al* [57] and Latva-Kokko and Rothman [6], using the method of Chapman-Enskog analysis. The dynamics accord with a simulation designed to address multi–component flow in the continuum approximation and provide greatly improved simulation performance as follows.

The segregation method analysed has been shown to produce a very marked reduction in the interfacial micro-current activity associated with a curved interface, which makes the method valuable (i) in terms of the improved quality of its results and (ii) by facilitating simulational access to regimes of flow with low capillary number and drop Reynolds number, see Chapter 7. Whilst it (the formulaic segregation method) produces an interface which is distributed in terms of its component index, the hydrodynamic boundary conditions which it enforces (when used in conjunction with Lishchuk's method [25]) support the notion of a sharp, unstructured, continuum interface as effectively as other LB methods. Furthermore, it is possible to use further algorithmic extensions with the model discussed here to improve performance in this respect. It is also worth noting that the distribution of the interface is not dependant on resolution, therefore a narrow interface may effectively be modelled with an increase in system resolution.

For continuum applications, it was found that, in addition to improved hydrody-

namic properties, a formulaic segregation method is considerably more computationally efficient than other methods (eg. numerical segregation described in Chapter 7), especially when used with a locally calculated interface normal, which the method is able to support under certain circumstances.

9 Kinematic Condition for Distributed Interface

In Chapter 7, initial attempts were made to impose a kinematic condition; these initial attempts were admittedly primitive and computationally slow; here is presented a, transferable, efficient and effective algorithmic enhancement designed to improve the accuracy of all multi–component lattice Boltzmann methods when applied to the simulation in the continuum approximation of fluid mechanics. The method presented in this chapter also benefits from a more well-founded theoretical basis.

By applying a collision parameter (kinematic viscosity) perturbation to reduce velocity gradients in the immediate interfacial region, a kinematic condition of mutual impenetrability between separated fluids is effectively enforced. Matters relating to a variation in the collision parameter are also briefly discussed.

For consistency with previous chapters, a brief introduction and background are presented here in an effort to make this chapter reasonably self contained. The familiar reader may wish to overlook these sections and continue directly to section 9.3.

9.1 Introduction

The simple and robust innovations outlined in this chapter are relevant to all MCLB variants when applied to the continuum regime, with completely immiscible fluids. The aim of this chapter is to address the representation of an interfacial kinematic condition (KC) of mutual fluid impenetrability [5]; in section 9.2 the most appropriate MCLB model for such an application will be described, then, in section 9.3, it is shown how the method may be simply and effectively modified explicitly to contain a KC in a manner readily transferable to other MCLBs; finally, in section 9.4 the results of this section of work shall be presented and then discussed.

Interfacial boundary conditions are treated in all MCLB as a constraint coupling of the dynamics of a single component lattice Boltzmann (LB) fluid to those of an

order parameter [22] or a phase field as in previous chapters and [25]. Thereafter, the principal MCLB methods are distinguished by the detailed way in which a fluid-fluid interface is imposed [3], [21, 25]. with different algorithms favouring different applications.

The MCLB method of Chapter 8 is used for this work and comprises a synthesis of the work of Lishchuk *et al* [25], d'Ortona *et al* [57] and Latva-Kokko and Rothman [58]. As shown in previous chapters it has narrow interfaces with a thickness independent of the computational mesh resolution (but see below), an independently adjustable interfacial tension which may be large, predictable phase field dynamics and it also facilitates simulation at low capillary and drop Reynolds number see chapters 7 and 8. Furthermore, the MCLB method of chapters 7 and 8 is based on the notion of dynamic, stress, boundary conditions [5] enforced over the interface between immiscible fluids [25] however neither it nor its progenitors take explicit account of any form of KC.

The present work goes some way to addressing the problems of conflicting length scales, unavoidable in MCLB, given that any practical (stable) interface must be diffuse on some characteristic, finite distance. We address this issue of interface scale and locatability implicitly, by devising a simple KC algorithm which acts over a very limited distance on either side of a defined, sub–grid interface centre, which, in turn, the KC implicitly serves to define.

9.2 Background

The MCLB method used in this chapter is that described in Chapter 8 which in itself is a continuation of work carried out in Chapter 7. It builds from a single component, single relaxation time LB variant designated the LBGK model [4].

For present purposes, the body force needs to contain spatial variation which necessitates spatial variation in $\phi_i(\mathbf{r})$ see section 7.4. This obstructs the emergence

of the lattice Navier Stokes equation (212). As stated previously, a solution to this problem, given by Guo *et al*, requires (i) a more complicated relationship between $\mathbf{F}(\mathbf{r})$ and $\phi_i(\mathbf{r})$ than that of equation (213) and (ii) a re–definition of $\mathbf{u}$. The modified relationships are stated in equations (219) and (220). Now, consider the form of fluid interface force which ϕ_i should generate, and where this force should be applied.

Utilising the same notation as that of Section 8.6, post–collision, post–segregation (re–coloured) quantities are indicated by the use of a double dagger superscript. Accordingly, the Latva-Kokko and Rothman's form of d'Ortona's segregation is written as [58], [57]:

$$R_i^{\dagger\dagger} = \frac{R}{R+B} f_i^{\dagger} + \beta \frac{RB}{R+B} t_p cos(\theta_f - \theta_i) \left|\mathbf{c}_i\right|, \tag{308}$$

in which, recall, θ_f (θ_i) is the angle of the colour field, $\nabla\rho^N$, (link) and $R_i^{\dagger\dagger}$ denotes the post–collision, post segregation value of the red fluid's momentum density associated with link i. Note that, for stable continuum interfaces segregation parameter, $\beta < 0.71$, see Chapter 8.

Equation (291) of Chapter 8 describes the dynamics for the phase field scalar, ρ^N for any MCLB scheme based upon the segregation method outlined above which uses a LBGK model and either the standard method of applying a body force, or Guo's more accurate method:

$$\partial_t \rho\rho^N + \nabla.\left(\rho\rho^N \mathbf{u}^*\right) = -f\nabla.\left(\rho^N \mathbf{F}\right) + \frac{1}{2}c_s^2 \nabla.\left(\rho^N \nabla\rho\right). \tag{309}$$

In fact, the order parameter employed in the Oxford MCLB mesoscale model [21] to define the component (or, more accurately in this case, phase) obeys a broadly similar dynamical equation.

For the standard model (with a direct velocity, $f = 1$ in equation (220)), as demonstrated in Chapter 8, it is possible to obtain for the dynamics of the phase

field:

$$\frac{D}{Dt}\rho^N = -\frac{1}{2\rho}\mathbf{F}.\nabla\rho^N, \tag{310}$$

which equation clearly deviates significantly from the correct form required for a continuum fluid, namely (see section 9.3):

$$\frac{D\rho^N}{Dt} = 0. \tag{311}$$

However, with Guo's more accurate representation of body force (with an indirect velocity, $f = 1/2$ in equation (220)), the phase field dynamics takes a more amenable form. From Chapter 8 recall that using equations (276), (288), (292) and (293) directly, it is possible to obtain:

$$\frac{D}{Dt}\rho^N = \frac{1}{2\rho}\nabla.\left(c_s^2\rho^N\nabla\rho - \rho^N\mathbf{F}\right). \tag{312}$$

We return to this equation in the next chapter.

9.3 Kinematic Condition

In a continuum fluid, the interface is subject to a kinematic condition (KC) which effectively requires that the interface move at the same speed as the local fluids. For a point in the interface, identified by a chosen value of ρ^N, to advect in flow $\frac{D\rho^N}{Dt} = 0$. It follows from equation (310) or (312) that the interface is accelerated relative to the local fluid by an amount determined by pressure and phase field gradients. This chapter is concerned with a remedy to this absence of an intrinsic KC, portable into other MCLB methods and rather more in accord with the spirit of LB simulation than earlier attempts based upon a momentum conserving process of averaging the separated fluids' velocity in the interfacial region see Chapter 7. The latter method is of very limited value; it is reliant upon arbitrary, computationally expensive site selection criteria and it compromises interface resolution.

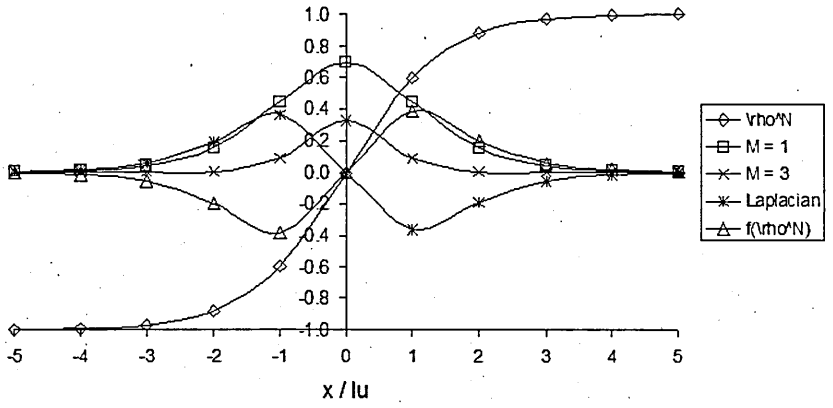

Figure 40: Variation of quantities across the interfacial region. In the figure key, $M = 1, 3$ identifies the variation of $\delta\omega/K$ in equation (313) for $M = 1, 3$, f denotes $f(\rho^N)$ in equation (317). The interface, defined by $\rho^N = 0$, is centred on $x = 0$ locally.

Consider a static, spherical red drop, nominal radius $r_0 = 50$, suspended in a blue fluid, such a drop is described by a phase field variation $\rho^N = tanh(\beta(r - r_0))$ for $\beta < 0.71$ as derived in Chapter 8. Figure 40 shows the variation of the phase field, ρ^N, in the radial direction, across the interface (diamonds); note, abscissa $x = 0$ is the interface centre. Variation of the Laplacian of the ρ^N field (which, broadly, determines the divergence of $\mathbf{F}$, which is the principal contribution to the right hand side of equation (312)) is plotted using asterisks. On contour $\rho^N = 0$ (at $x = 0$) of this particular MCLB's interface, a KC is approximately satisfied in both models. In general, however, the gradients of ρ^N vary and the interface is accelerated most rapidly for positions $|x| \leq 1$, corresponding to $|\rho^N| < 0.5$. This is true for all interface radii or curvatures. The region $|\rho^N| < 0.5$ suggests itself as the interfacial zone onto which algorithmic extensions designed to promote a KC should be concentrated.

To promote correlated motion between the internal (red) and external (blue) fluids at the boundary we seek to eliminate fluid shear in a thin shell $|\rho^N| < 0.5$, which, from figure 40, is approximately 3 lattice units thick. This is accomplished

by increasing the local sum fluid viscosity using a collision parameter perturbation:

$$\delta\omega = -K\left(1-\rho^{N2}\right)^{M}, \qquad \left|\rho^{N}\right| \leq 0.5, \qquad K, M > 0, \tag{313}$$

corresponding, in D2Q9, to a local viscosity increase:

$$\delta\nu = \frac{1}{3}\frac{K}{\omega^2}\left(1-\rho^{N2}\right)^{M} = \frac{1}{3}\frac{K\beta^{M}}{\omega^2}\left|\nabla\rho^{N}\right|^{M}, \tag{314}$$

in which use has been made of a property of interfaces generated by the formulaic segregation of equation (308), after d'Ortona [57] and Latva-Kokko *et al* [58], derived in Chapter 8, namely:

$$\left|\nabla\rho^{N}\right| = \beta\left(1-\rho^{N2}\right). \tag{315}$$

The requirement $\nu > 0; 0 < \omega < 2$ constrains the choice of K; figure 40 plots the variation of $\delta\omega$ for $K = 1$, $M = 1$ (open squares) and $K = 1$, $M = 3$ (diagonal crosses); for the case of $M = 3$ perturbation $\delta\omega$ is significant only for $|\rho^{N}| > 0.5$, which advantageously limits any penalty on resolution in the interfacial region; with $M = 3$ sites at $x = \pm 2$ are unaffected, further adjustment of $M : M > 3, M < 10$ does not change this observation but degrades stability. Clearly, the extension in (314) generalizes readily to all other MCLB variants [22], [33], and is easy to apply in 3D.

When considering incompressible liquids, one traditionally factorizes $\nu(\omega)$ in the lattice Navier–Stokes equation (212), whereupon any spatial variation in the latter will generate additional force terms. This encourages the view of an incompressible, fluid, interfacial region of variable viscosity, subject to a compensatory forcing which effectively restores a KC. It is worth remarking that a variation in kinematic viscosity, ν may characterize a particular problem, for example when modelling spatial variation in fluid temperature within the Boussinesq approximation; where thermo-hydrodynamic flow is modelled (i)by neglecting density variation with pressure but

allowing the pressure to vary with temperature and (ii) allowing the kinematic viscosity ν to vary with temperature. Any such variation should be modelled with ν, positioned as in equation (212), behind the first spatial differentiation in the second term in the right hand side of equation (212).

Consider the case of Guo's (adjusted velocity) model. Factorization of $\nu(\omega)$ exposes an effective, shear–rate dependant forcing in the right hand side of the lattice Navier-Stokes equation (221):

$$F'_\alpha = 2\rho S^*_{\alpha\beta}\frac{\partial}{\partial x_\beta}\nu(\omega) = -\frac{2}{3}\frac{\rho}{\omega^2}S^*_{\alpha\beta}\frac{\partial}{\partial x_\beta}\delta\omega, \tag{316}$$

in which the last equality uses a substitution for $\nu(\omega)$ for the case of a D2Q9 model. Here strain rate, $S^*_{\alpha\beta}$, is based upon gradients in the corrected velocity of equation (220), note.

With the proposed ω-variation of equation (313) it is possible to obtain from equation (316), by straightforward use of equations (314) and (315):

$$F'_\alpha = -\frac{4\rho\beta}{3\omega^2}KMf(\rho^N)S^*_{\alpha\beta}f'_\beta, \qquad f(\rho^N) \equiv \rho^N\left(1-\rho^{N2}\right). \tag{317}$$

It is not surprising to find an equivalence between an interfacial viscosity perturbation, suggested in equation (314) and an effective shear rate dependant force. The factor $f(\rho^N)$, defined above, varies across the interfacial region as illustrated in figure 40.

9.4 Results and conclusion

Figures 41 and 42 derive from data obtained from drops of initial radius 15 lattice units, with surface tension parameter $\alpha = 7.0 \times 10^{-3}$. For the data of figure 41 the drop was exposed to an unperturbed shear rate 1.0×10^{-4} in lattice units, on a lattice of size 150×50 lattice units; the unperturbed collision parameter $\omega = 1.0$ and

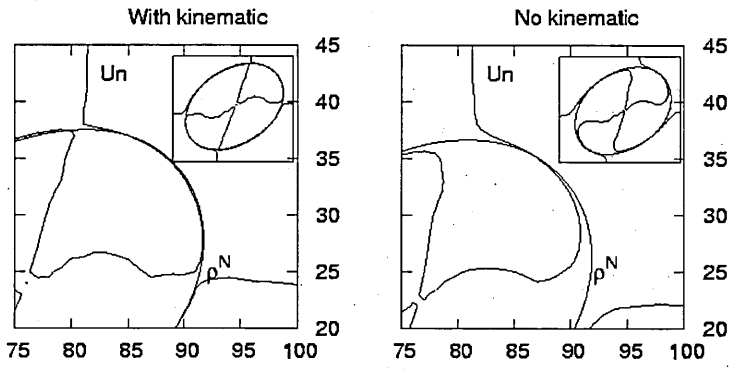

Figure 41: Normal velocity $u_n = 0$ contour superimposed over $\rho^N = 0$ contour of a sheared drop at steady state, with (left) and without (right) the kinematic condition of equation (313) in effect. Note the departure of the $\rho^N = 0$ and $u_n = 0$ contours in the 'polar' and 'equatorial' regions in the figure on the right.

the perturbation $\delta\omega$ (equation (313)) was characterized by $K = 0.5$, $M = 2$. For figure 42, the same lattice with rest boundaries was used, with a range of viscosity perturbations (see figure key and caption).

Consider a neutrally buoyant (red) drop embedded in a symmetrical sheared (blue) fluid of identical kinematic viscosity. At the steady state, in the rest frame of the drop interface deformation ceases and the normal component of fluid velocity, u_n, vanishes (this is untrue of the corresponding tangential component, which is non-zero in general). On noting that the chosen interface centre, the contour $\rho^N = 0$, cannot intersect any lattice node, it is suggested that the best illustration of the sub-grid nature of the KC and the benefits attending its use is the correlation between the contour $\rho^N = 0$ and the contour $u_n = 0$. Compare now the results in figure 41. On the right of this figure is shown detail of the flow in the region of the drop pole; the $u_n = 0$ contour is superposed over the (closed) contour ρ^N, for a drop without a KC inducing perturbation in force; the figure inset shows the whole drop. On the left of figure 41 is the corresponding simulation with a KC, characterized by $K = 1, M = 3$ applied. In the latter, the $u_n = 0$ contour is located much closer to the centre of the interface, especially in the polar region, where u_t, the component

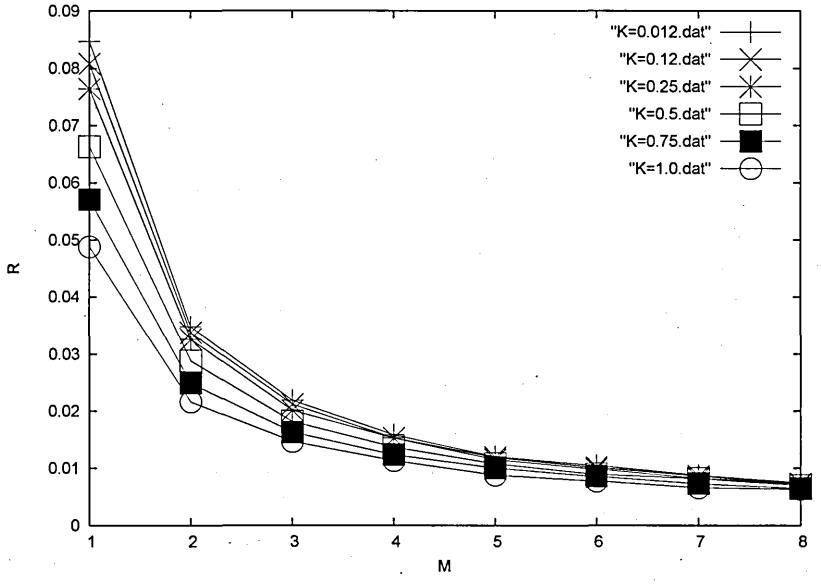

Figure 42: Absolute value of R, the steady–state residual of the interface normal velocity, u_n (ordinate), defined in equation (318), plotted against M (abscissa) for different values of K; M, K as in collision parameter perturbation equation (313) for the simulation defined in the first paragraph of section 9.4

of fluid velocity tangent to the $\rho^N = 0$ was also much smaller.

Briefly consider the effective, shear dependant force $\mathbf{F}'$ of equation (316) at the 'North Pole' of the drop depicted in figure 41. The shear rate at the drop North Pole is dominated by a contribution $\frac{\partial v_x}{\partial y} > 0$; this, and the variation of the factor $f(\rho^N)$ imply $F'_x < 0$ ($F'_x > 0$) for the blue (red) side of the interface, in accord with a reduction of the local shear gradient when $\rho^N = 0$.

Figure 42 considers the correlation of the zero of normal velocity with the $\rho^N = 0$ contour. The ordinate corresponds to a steady-state residual of the modulus of the normal velocity:

$$R = \sum_{\rho^N=0} |u_n| \,, \tag{318}$$

the abscissa to the value of parameter M in equation (313). Branches of the plot with increasing ordinal intercept correspond to amplitude parameter K= 1.00, 0.75, 0.50, 0.25, 0.12, 0.012 in equation (313). Simulations show that the correlation

improvements at a diminishing rate as M increases; whilst our simulations were stable, as M increases larger shear rates may cause instability.

The segregation method outlined briefly in this chapter as a reiteration of the method described in detail in Chapter 8, when used in conjunction with a kinematic condition facilitate robust MCLB simulations in the continuum approximation with advantages of increased efficiency, negligible interfacial micro-current activity and drop pinning [58], Chapter 8 (facilitating low capillary and drop Reynolds number applications), improved representation of continuum hydrodynamic boundary conditions (figure 41) and, not least, simplicity. Moreover, the simple rule expressed in equation (313) transplants into other MCLB methods [21], [33] directly, to facilitate their application to the continuum regime.

10 Wetting with continuum lB : *Simulation of 2D dense Films Under Gravity*

10.1 Introduction

The work carried out in this chapter combines and relates the work described previously, on the forced mass conserving boundary condition in Chapter 6, the distributed interface of D'Ortona *et al* [57] and the kinematic condition discussed in the previous chapter 9. The focus of this chapter is to produce an accurate and robust continuum scale wetting condition.

Clearly the wetting problem is very long-standing and is of great importance in a range of situations; from medical applications, where leukocytes have been modelled as wetting drops, to industrial processes, such as paints and coating applications, to the new science of micro–fluidic devices (see also section 2.1 for recent advances).

It is important to stress at this stage that the aim of this chapter is not to produce a microscopic wetting theory. State of the art research in this area combines non–equilibrium molecular dynamic calculations (which operate on very short length and time scales in the near–wall region) which are appropriately coupled to continuum flow solvers, this allows the solvers to cross the length scales between the surface slip, which is molecular in origin, and the continuum scale flow that this slip determines. In fact there are a range of more or less phenomenological methods that have been used to describe slip and wetting in continuum calculations. The dynamics and spreading of liquids has been treated in considerable depth [62] and references therein. Most of these methods attempt to use the stresses at the boundary to predict a slip velocity. This qualitative but secure and well accepted fact is our principle modelling assumption. Accordingly the work undertaken in this chapter focuses on an accurate, artefact free adaptation of *given* continuum boundary conditions to a multi–component lattice Boltzmann equation flow calculation.

We begin by reminding the reader that, as discussed in chapter 4, when considering a solid boundary with a no-slip boundary condition applied, the relative motion of a dynamic contact line (DCL henceforth) between the red and blue fluids along the boundary will result in a singularity. We shall assume one, existing, model of slip at the contact point – the so–called Navier slip condition, (as described in section 4.4). Again, our model is not intended effectively to provide a theory of slip velocity *on the fly*; rather it adapts lattice Boltzmann simulation in a flexible, robust manner, without any need for additional assumptions beyond that of a given theory of slip. In overview, this is accomplished as follows.

At the contact point we (i) specify the wetting angle as a boundary condition on the interfacial normal: by (ii) devising an appropriate mass and species conserving boundary condition which (iii) uses the second–order accurate closure (as set–out in Chapter 6) to specify, for a measured distribution of fluid stresses, an instantaneous contact–point slip velocity. Our results are obtained using an assumption of a quasi–static slip *ie.* we will assume that slip velocity depends only upon the instantaneous distribution of fluid stress (an idea that was initially introduced in Chapter 4) according to:

$$v_s = b\frac{\partial u_x}{\partial y}, \tag{319}$$

were the parameter "b" is known as the slip length and is a molecular property of the fluid, with, we assume, similar status to the kinematic viscosity.

10.2 Background

As mentioned previously predominantly in Chapter 7, at large flow velocities the effects of the micro–current is negligible, however, when the velocity of the flow becomes comparable to the velocities generated by the interface algorithm, significant anomalies in the method begin to appear. The motion of a DCL might be very slow,

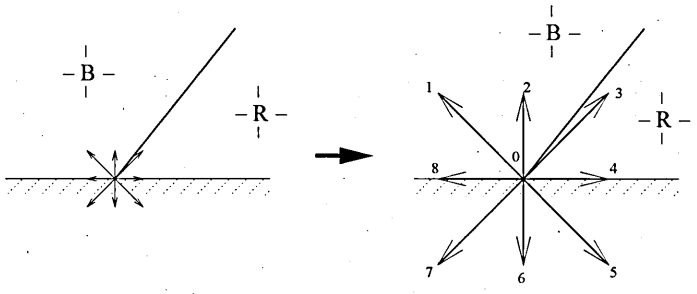

Figure 43: D2Q9 lattice site positioned at the dynamic contact line (DCL) on the bottom surface of a flat duct (left) and enlarged (right). The heavy line shows the supposed location of the interface separating red (R) and blue (B) fluids.

so any micro-currents developed by the interface algorithm at the DCL will effect local flow velocity and flow velocity gradients and thus the motion of the contact itself and the dynamics of the flow. Put another way, contact point slip at the boundary must be assumed to involve small motions relative to the boundary, the presence of micro-currents will interfere with the detail of the slip model and resulting flow: this underscores the need for a low micro–current interface algorithm, such as that developed in Chapter 8. It should also be clear that, in addition to minimisation of micro-currents there must be exact conservation of mass and colour; there cannot be differential loss of either colour species close to the contact point.

We aim here to specify the fluid behaviour in terms of a chosen wetting model; a known contact angle or wetting angle, θ_w, that is taken to be a molecular property with the same status as viscosity and, of course, the surface tension.

For the sake of continuity with previous chapters (chapters the analysis contained within this chapter will consider a D2Q9 node assumed to be situated at the bottom of a flat duct flow as shown in Figure 43. This node is situated in the boundary between the red and blue fluids and so its links contain non–zero momentum distribution function of both components, R_i and B_i. The direction of the x-axis is taken to be tangent to the boundary.

The wetting angle, θ_w, as mentioned above, is related to the colour field direction as illustrated in figure 44 and may be expressed as a boundary condition on the

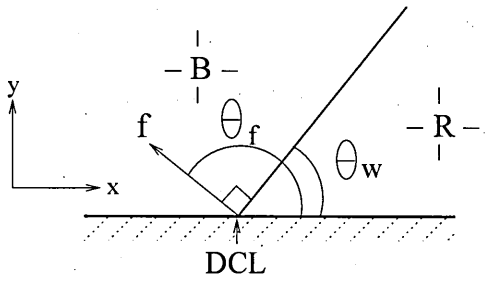

Figure 44: Boundary condition on the colour field direction at the DCL; the direction of the colour field is defined in relation to the wetting angle for wetting at the bottom of a duct. See equation (320) and fig 43

colour field direction θ_f as :

$$\theta_f = (\theta_w + 90), \quad x = 0. \tag{320}$$

The boundary condition associated with red and blue species is essentially one of mass conservation and is illustrated in fig 43.

Similarly, the fluid velocity on the boundary is predicted by the following slip law and kinematic condition:

$$u_x(y=0) = b\frac{\partial u_x}{\partial y}, \quad v_y(y=0) = 0, \tag{321}$$

in which $\frac{\partial v_x}{\partial y}$ is measured, note, we return to this issue in section 10.3.

It is appropriate here to review the principal features of the boundary closure method that is now to be adapted to the situation of slip.

Notwithstanding detailed points revised below, the method of Chapter 6 essentially requires that (i) a boundary, or wall, velocity distribution, $\mathbf{u}_w$, (ii) an effective density, ρ_w (not the same as the mass, M_{in} arriving on uncut links of the boundary site, note) and (iii) that the wall fluid strain rate, $S_{\alpha\beta w}$ henceforth, be known or measured. Uncut boundary links' momentum distribution function, f_i, is then considered to be given by the sum of its $f_i^{(0)}(\rho_w, \mathbf{u}_W)$'s and $f_i^{(1)}(S_{\alpha\beta w})$'s. It is important to recall that, in the method of chapter 6, we choose to make calculated $f_i^{(0)}$'s ($f_i^{(1)}$'s)

to sum to M_{in} (zero) on uncut links.

To apply this method in the present context we need therefore to calculate or measure wall densities and wall strain rates. The latter are doubly important now as they're used to find the wall velocity through the assumed slip model. As we shall see below, the problem of finding the wall density, ρ_w is complicated by the need to apply a (variable) external force for interfacial tension and the problem of measuring the velocity gradient $S_{\alpha\beta w}$ is complicated by the fact that the evolving wall velocity, u_w, is not known and that the previous use of spatial translational invariance in Chapter 6 is lost in contact region when a DCL is considered. The simplest solution to this problem is that outlined in section 10.3.

In the method of Chapter 6 the effective wall density, ρ_w, was defined such that the total mass on post–collision uncut links is equal to the mass, M_{in}, which propagated onto the wall node links at the beginning of evolution step. It was also seen that the source term does not conserve mass on uncut links. Appropriate calculation of the effective wall density, ρ_w which takes account of these facts and the fact that there are now two different, individually conserved fluids present at the boundary is set out in section 10.4. In section 10.4 we will deal with the calculation of wall strain rates and slip velocity and assignment of corresponding $f_i^{(1)}$'s. Section 10.4.1 will deal with re–colour.

As stated in the initial section of 10.2, it is important in this work to maintain the low velocity capabilities of the algorithm. Should the wetting procedure induce spurious velocities that are comparable to the velocity of flow itself, the calculated dynamics of the flow will be corrupted.

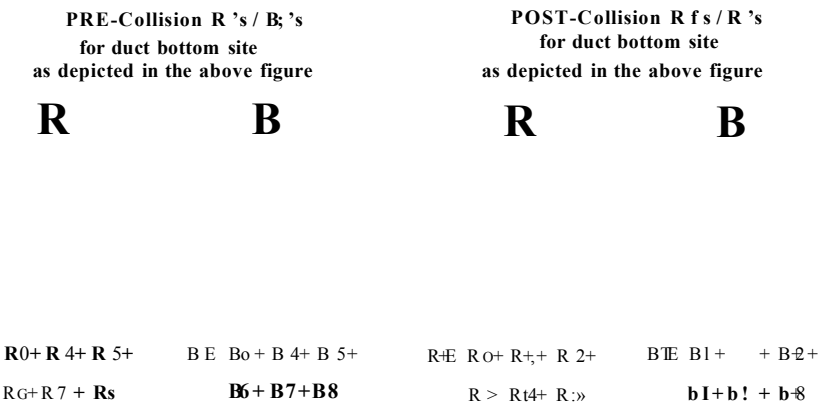

Figure 45: Colour conservation boundary condition for the D2Q9 lattice site depicted above. The dotted line indicates the position of the boundary; empty circles indicate the absence of information or locations where link information is not required.

10.3 General Remarks on the Implementation of the Slip Condition

We clarify at the outset that the particular means by which boundary condition (321) is applied in the LB method is reported in this chapter. At location x on the boundary $y = 0$, when measuring the gradient ^ with say a fourth order correct finite difference we have;

$$ux(x, y = 0) = b\{a3ux(x, y = 3) + a2ux(x:y = 2) + aiux(x, y = 1) + a0ux(x, y = 0)\}, \tag{322}$$

where a3...a0 are the appropriate coefficients for the finite difference form from which the slip velocity $ux(x,y = 0)$ is easily determined;

$$ux(x, y = 0) = \frac{azux(x\,ly = 3) + a2ux(x,y = 2) + aiux(x,y = l)}{(1 - a0)}. \tag{323}$$

10.4 Application of Formulaic Segregation at a Forced Boundary Node with Slip

The situation is now complicated beyond that discussed in Chapter 6 by the fact that there are two fluids present that need to be conserved individually. As with the situation described in Chapter 6, there is once again an external force applied to the continuum fluid which is specified and therefore known. The external interface inducing force must be consistent with the wetting angle, θ_w, see fig 44. The gradient of the phase field, $||\nabla\rho^N||$ and the curvature, k, are determined using finite differences. As we know all of the aforementioned quantities, we can calculate the overall wall force $\underline{F}$. The associated mass "defect" may then be calculated from the following expression:

$$\Delta M = \sum_{i\neq 5,6,7} \phi_i, \quad \phi_i \equiv t_p \left(1 - \frac{1}{2\tau}\right) \left(3\left(\mathbf{c}_i - \mathbf{u}^*\right) + 9\left(\mathbf{c}_i.\mathbf{u}^*\right)\mathbf{c}_i\right).\mathbf{F}(\mathbf{r}). \tag{324}$$

ΔM is the mass effectively removed or introduced to live or uncut links by the action of the surface tension force.

It is important to note that the mass defect, ΔM will be eventually corrected for live or uncut links when the source term ϕ_i is added.

For the boundary node depicted in figures 43 and 44, the incoming masses are calculated from:

$$M_{in} = \sum_{i\neq 1,2,3} f_i, \tag{325}$$

$$R_{in} = \sum_{i\neq 1,2,3} R_i, \tag{326}$$

$$B_{in} = \sum_{i \neq 1,2,3} B_i, \tag{327}$$

and we define;

$$M'_{in} = M_{in} - \Delta M. \tag{328}$$

10.4.1 Effective Red and Blue Fluid Densities for Boundary Node

In order to apply the formulaic segregation of D'Ortona *et al* [57] (which was discussed in detail in Chapter 8) *without any adjustment* we need the effective red and blue densities, R' and B', as well as M'_{in}. By way of definition R' and B' should be such that application of the formula;

$$R_i'^{\dagger\dagger} = \frac{R'}{R'+B'} f_i^{\dagger\dagger} + \beta t_p \frac{R'B'}{R'+B'} cos(\theta_i - \theta_f).|\mathbf{c_i}|, \tag{329}$$

will produce $R_i^{\dagger\dagger}$'s and $B_i^{\dagger\dagger}$'s which total to R_{in} and B_{in} according to equations (326) and (327), on uncut links. Colour conservation requires that:

$$\sum_{i \neq 5,6,7} R_i^{\dagger\dagger} = R, \tag{330}$$

$$\sum_{i \neq 5,6,7} B_i^{\dagger\dagger} = B, \tag{331}$$

where, after the notation of Chapter 8, a double dagger superscript denotes a post-collision post-recolour value.

Now let, $R'(B')$ denote an effective value of $R(B)$ such that :

$$\sum_{i \neq 5,6,7} R_i'^{\dagger\dagger} \equiv \sum_{i \neq 5,6,7} \left(\frac{R'}{R'+B'} f_i^{\dagger\dagger} + \beta t_p \frac{R'B'}{R'+B'} c_i \widehat{f} \right) = R, \tag{332}$$

where we have used equation (329).

Note that it is $f_i^{\dagger\dagger}$ and not $f_i^{\dagger}$ in the above equation; $f_i^{\dagger\dagger}$ includes the source

contribution ϕ_i so, overall mass conservation requires that :

$$\sum_{i\neq 5,6,7} f_i^{\dagger} = M'_{in}, \tag{333}$$

where, $M'_{in} = (M_{in} - \Delta M)$, recall.

Now, the factor $\beta \sum_{i\neq 5,6,7} t_p f_i c_i$ in the second term of equation (332) is a function of θ_w and particular boundary node geometry only. It is therefore straightforward to show that the following must be true for a node at the bottom of a channel (situation depicted in figures 43 and 44):

$$\beta \sum_{i\neq 5,6,7} t_p f_i^{\dagger\dagger} c_i = \frac{\beta}{6}.f_y, \tag{334}$$

where f_y is the y component of the boundary colour field.

It is then possible to obtain the following relationship from equation (332) :

$$\frac{R'}{R'+B'} M_{in} + \frac{R'B'}{R'+B'}.\frac{\beta}{6} f_y = R. \tag{335}$$

On using the effective density identified in Chapter 7 we must clearly have:

$$\rho' = R' + B'. \tag{336}$$

Equation (335) now becomes :

$$\frac{R'}{\rho'} M_{in} + \frac{R'(\rho' - R')}{\rho'}.\frac{\beta}{6} f_y = R, \tag{337}$$

which re-arranges straightforwardly into a quadratic for the effective red density, R', for the wall site of the node depicted in fig 44:

$$(\beta f_y)\, R'^2 - \rho'(5 + \beta f_y) R' + 6R\rho' = 0, \tag{338}$$

where the fact that $\frac{M_{in}}{\rho'} = \frac{5}{6}$ has been used, as was initially done in Chapter 7. Equation (336) may also be used to determine the effective blue density B'.

It is now possible to solve the above equation for R' to find the effective red and blue masses on a boundary site. Boundary site re–colour (or segregation) procedures are then carried out using this fictitious value, that is, using formula (329). Put another way, by using R' from equation (338) re–colouring may be carried out exactly as described in the previous section *without any further modification*, resulting in mass conservation and colour mass conservation on uncut links. The boundary node, including cut links, mass and colour are not conserved but clearly this is of no consequence. Note that the quadratic (338) must be solved at every boundary site as R' varies through the contact region.

Recall that in Chapter 6, third order forward and backward finite difference methods were employed to measure the fluid stress to solve the system of equations (152) for example. Due to the axial symmetry of the system considered in Chapter 6, many of the stress derivatives in equations (152) were constrained to zero. The presence of a DCL now violates the symmetry of the previous system and this assumption is no longer valid. All derivatives must now be calculated. The method of calculation of $f_i^{(0)}(\rho, \underline{u}_o)$ and $f_i^{(1)}$ however, being third order forward and backwards difference methods, remains the same as in Chapter 6 and Chapter **??** with the following simple but important change.

10.5 Stress Measurements with Spatially Variable Force

Here we make an important revision to the results and methods of Chapter 6.

In the Guo LBGK model used in this work to impress a variable body force, the Chapman-Enscog analysis works differently. By straightforward but tedious algebra, it is possible to extract from the reference [48] the modified relations for the zeroth to second moments of the higher components of the pre-collision distribution

function, with the lattice basis. That is, when using the scheme for spatially variable forces $\underline{F}$, used in chapters 7 and 8, there arise important changes from the usual Chapman-Enscog analysis of Chapter 4, which underpins the work of chapters 5 and 6. We must take account of these now, before we use the results of chapters 5 and 6. As we have just said, tedious but straightforward algebra allows us to extract from [48] the following modified relations for the zeroth to second order moments of the higher components of the pre-collision distribution function when we have a spatially variable force.

$$\sum_i f_i^{(1)} = 0, \tag{339}$$

$$\sum_i f_i^{(1)} c_{i\alpha} = 0, \tag{340}$$

$$\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta} = -2\rho c_s^2 \tau S_{\alpha\beta} - \frac{1}{2}\delta_t \left(F_\alpha u_\beta + F_\beta u_\alpha\right) \longrightarrow -2\rho c_s^2 \tau S_{\alpha\beta}. \tag{341}$$

Where we have replaced the strain rate as follows:

$$S_{\alpha\beta} \longrightarrow S_{\alpha\beta} + \frac{1}{4\rho c_s^2 \tau}\delta_t \left(\overline{F_\alpha}\overline{u_\beta} + \overline{F_\beta}\overline{u_\alpha}\right). \tag{342}$$

Therefore, the methods of Chapter 6 can be easily applied to the problem discussed here but only after we replace the measured stress in e.g equation (152) with values calculated using the replacement 342.

10.6 Results

In order to establish weather or not the wetting algorithm induces spurious velocities, a static drop at the bottom of a duct is simulated at steady state. From figure 46

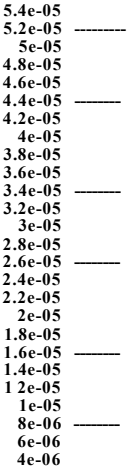

Figure 46: Velocity field for a static drop at steady state. Note the micro current activity close to the contact point on the boundary.

it is possible to see that, at steady state, the spurious velocities that exist at the contact region remain small and comparable in magnitude to those generated within the bulk scheme. From this result it is evident that the low flow velocity capabilities of the algorithm are retained near the boundary.

For the continuum regime calculations, the wetting angle and the slip length, b, for example are required to determine the hydrodynamics. The thickness of the interface and the detail of the algorithm should *not* effect the observed wetting dynamics. Results shown in this section do not change with adjustment of the *f3* parameter for values $0.5 < (3 < 0.7$.

Figure 46 shows detail of the micro-current in the vicinity of the contact for an attached drop at steady state; superposed contours show constant w2, the highest velocity observed in the contact region is 0(10“5). This is very small when compared to, for example, the Oxford algorithm.

Figure 47 is a summary of the dynamics of an attaching drop. The figure shows

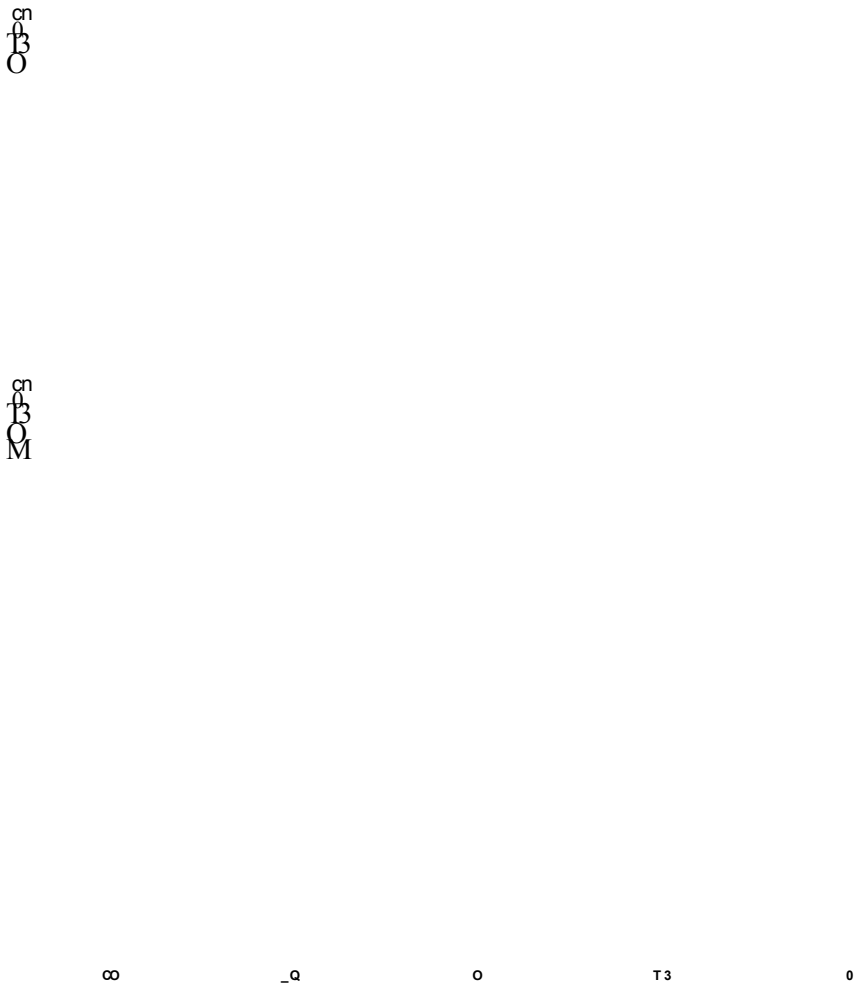

Figure 47: A drop of radius 22 lattice units in a channel of 150 lattice sites by 70 lattice sites with wetting, Dirichlet boundary conditions applied on $Y = 0$ and $Y = YLENGTH$ boundaries with periodic conditions left to right. Gravity acts in the negative y direction (where the x axis runs positively up the page and the y axis runs from right to left on the page). Images show wetting behaviour with varying wetting angles specified. "Snap shots" were taken at <7)10. 000/,, 6)30,000/;, c)50,000Z, *d)* 130,000/ and 250,000/. Tht^aram eter *j3* is set to 0.7 throughout

Figure 48: Detail of settled drop on the bottom surface of a channel of dimensions 400 by 90 (figure shows 150 to 300 of x-range and 0-50 of y-range). Drop has a specified wetting angle of 100° and a uniform body force of $a = 5 \times 10$;). Kinematic condition is applied and a slip length of 0.05 is specified.

the effects of a varying wetting angle on a 2D red drop that is initialised close to a flat boundary. The figure shows the evolution as the drop is released and approaches steady state with the slip length, b, set to zero. The effects of the slip length on these results was not investigated due to time constraints. The effects of the slip length are, however, considered in the following, more interesting sets of results which relate to simple (prototype) coating flows.

The result for film wetting in the presence of a constant body force applied from left to right to simulate gravity is shown in figure 48. The instantaneous stream function is calculated in the rest frame of the red fluid from:

$$\hat{}(x, y) = \int_{Jo} (ux(x,y') - U\,x0)dy', \tag{343}$$

where *Uxo* is the average velocity of the red fluid *which is not zero.* The rolling nature of the flow in the contact region is apparent in the structure of the stream-function contours superposed on the phase field information. Note that the kinematic condition of Chapter 9 is in force. Also note that the fact that the stream function contours intersect the phase field boundary does not imply that fluids are flowing across the boundary as the film is still spreading in all the images presented. The rolling nature of the flow is entirely emergent given a set slip length parameter

Figure 49: Identical simulation system as that shown in fig 48 however the entire simulation space is shown. The wetting angle in the above is set to 60° giving entirely different hydrodynamics to those emergent in fig48.

Figure 50: Identical simulation system to that depicted in both 48 and 49; again with the entire simulation space shown. The wetting angle is now set to 5°. The resulting hydrodynamics are again totally different to those shown in fig 48 and fig-49 above.

of 0.05 and the balance of the surface tension and the contact angle, no other forces or quantities have been set. The data for figure 48 was obtained with a wetting angle of 100°.

Figure 49 shows an identical system to that shown in 48 with the wetting angle changed from 100° to 60°. The flow in this case is seen to be qualitatively different to that shown in 48 however the flow is again totally emergent. The flow system depicted in figure 49 is what we consider to be a bifuricating contact; the red fluid that is in contact with the boundary flows in a clockwise motion rolling onto the surface while the red fluid adjacent to the interfacial region moves anticlockwise as it travels through the blue fluid.

Figure 50 depicts an exact replica situation to that shown in fig 48 and 49 however the wetting angle has been set to 5°. The rolling flow at the trailing edge of the drop is now lost. The flow structure is again totally emergent however the structure of the flow is now fundamentally different.

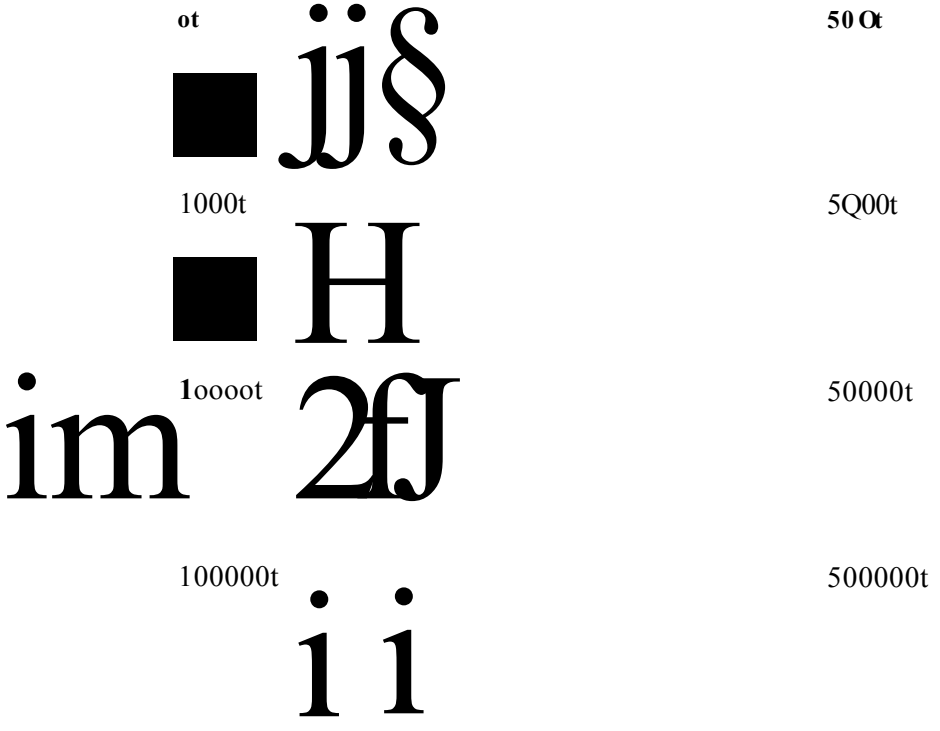

Figure 51: Red drop of radius 22 lattice units initialised in the centre of a channel of length 150 lattice units and a width of 45 lattice units, a uniform body force is applied from left to right. Navier slip and kinematic conditions are applied. A wetting angle of 60° has been specified.

Figure 51 depicts a channel of width 45 lattice units and a length of 150 lattice units, a drop is initialised in the centre of the channel with a radius of 22 lattice units. The preferred wetting angle at the top and bottom surfaces is specified at 60^o. The drop is seen to attach to top and bottom surfaces during the initialisation stage. Following initialisation, a uniform body force is applied from left to right. The drop then coats the surface of the channel drawing red mass from the central region of the drop and depositing it on the channel surface. If left to evolve, the coating effect completely drains the red fluid from the centre of the channel and the result is coating of the top and bottom boundaries completely. Again this behaviour is completely emergent.

10.7 Summary

The results which have been presented in this chapter represent a successful synthesis of work carried out previously and presented in chapters 6 – 9 inclusively. The formulaic segregation method of D'Ortona *et al* [57] has been found to be amenable to the adjustments necessary to make a very general boundary condition. The results all show that the induced micro-currents in the interface contact region are low (when compared with the bulk algorithm). By using a range of slip lengths and wetting angles the method has been shown to be capable of recovering various wetting and coating flows. In particular there is a qualitative, fundamental difference in the structure of the flows in the contact region between figures 48, 49 and 50 and we suggest that the range of bifuricating and rolling flows recovered show that our core MCLB method can be used to simulate DCL behaviour with confidence. We must be cautious about interpreting the data too closely. Time constraints permitted only **qualitative** evaluations of the closure method of this chapter. It is far from safe to attempt, for example, to differentiate between situations where rolling and bifuricating flow in the contact might occur.

Whilst time constraints prevented the generation of further results, those that are presented here clearly demonstrate the potential and flexibility of the method.

Further discussion of these results is to be found in the following chapter where future work is considered.

11 Prospective Future Work

11.1 Introduction

Although numerous advances to the multi–component LBGK method have been presented in this work; there are many points that at which it is possible to pursue future developments.

In this section I intend to outline what I perceive to be the most substantial problem with the algorithm as it stands, I also intend to propose a possible solution, which due to time constraints, has not been explored further than the suggestions I am about to make.

11.2 Curvature Discontinuity

At steady state, with zero flow in the far field, the wetting algorithm produces, under certain circumstances, spurious velocities (that are depicted in figure 52, where the stream function is superimposed on a plot of the ρ^N field). This is not surprising and all numerical methods must, at some level, produce artefacts. In our case the obvious anomalies in the algorithm produce vortices apparent in the stream function for example figure 52. This anomaly is important as it limits the application of the method to wetting problems where the method is most useful. Put another way, it bounds its domain of applicability.

From my data it is apparent that a spurious behaviour originates at the interface contact points. It has been found that our curvature calculation breaks down at the DCL for prolonged run-times (Recall, the MCLB interface algorithm evaluated, numerically, the interface curvature). Plotting the measured curvature data for the system shown in figure 52 shows that there is an obvious anomaly at the DCL at long simulation flow times, in steady flow. This anomaly, we believe, accumulates a flow artefact. Again this is not surprising as k cannot be defined where there is a

Figure 52: Flow defect occurring for prolonged run times superimposed 011 the phase field information. Simulation shown is for a drop of wetting angle 60° with a uniform body force applied from left to right.

discontinuity in /- the very definition of the DCL. The spurious values of curvature shown in figure 53 reach values in the region of 1.59 at their maximum in the DCL contact region.

Recalling the curvature calculation method employed in this algorithm initially introduced in Chapter 7:

$$= \quad n^{*}n^{*} \quad \backslash d\,y\,n^{*} \quad + \quad o\,i\,Uy\sim$$

The issue is not simply one of numerics; it can be seen in equation (344) that the curvature is dependant on gradients in the interface normal. At the DCL, the interface ceases to exist and therefore it's gradients cease to exist, resulting in a discontinuity in the curvature calculation as shown in figure 53. For a closed interface there is no issue as the interface is continuous everywhere but in the case of a wetting drop, the interface must stop somewhere.

In the present work the problem is partially avoided by use of appropriate difference methods to calculate the curvature at boundary nodes by extrapolation of the curvature at near boundary nodes. This expedient has allowed the current results to be obtained. More work and a more painstaking treatment of the curvature at the DCL might well resolve this issue, however there is a wider issue.

Figure 53: Curvature plot superimposed on the phase field for the system shown in figure 52.

11.3 Possible Solution

To provide a more "stable" solution to the issue of curvature above independent of the assumptions made about the treatment of DCLs at a boundary, we recommend a method of interface forcing which does not require a measurement of curvature.

Work on the problem is currently proceeding along the following lines:

The stress perturbation;

$$\mathrm{A}\,SaP \ll u\,f\,af\,0t \tag{345}$$

(where a; is a weight function) can be shown to have the following effect on the interfacial stress contractions in the interfacial regions:

$$\mathrm{A}\,Sapnp = u[n\backslash p, \tag{346}$$

$$\mathrm{A}\,Scfftff = 0, \tag{347}$$

where np and tp areinterface normal and tangent respectively. It can be shown that the above perturbation(345) is consistent with the followingeffectiveinterface forces at the macroscopic level;

$$Fa = dpASap, \tag{348}$$

which, when used in place of the interface producing force used in this thesis, namely;

$$F_\beta = \frac{3}{2} k\alpha \partial\beta \rho^N, \tag{349}$$

avoids the use of a measure of curvature - in the bulk algorithm and at the boundary. Put another way we can show that the body force:-

$$F_\alpha = \partial_\beta \Delta S_{\alpha\beta}, \quad \Delta S_{\alpha\beta} = \omega f_\alpha f_\beta, \tag{350}$$

this generates an interfacial pressure step in accord with the Laplace Law, and also appropriate stress contractions in the interfacial region, *without any further need to measure local curvature.*

The above method has been found to produce an interfacial tension without the requirement to calculate curvature using equation (344). The method however remains in its infancy and currently is producing significantly larger micro-current signals than the previous curvature calculation method. However it is possible to switch between either method and therein lies a possibility; one might utilise the curvature based, low micro-current method for the bulk interface region and the modified forcing method albeit with higher micro-currents for the boundary region. This should give the basis for a more artefact free wetting algorithm.

References

[1] B. Hasslacher U. Frisch and Y. Pomeau. Lattice-gas automata for the navier-stokes equation. *Phys. Rev.Lett*, 56:1505–1508, 1986.

[2] G. McNamara and G. Zanetti. Use of the boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett*, 61:2332–2335, 1988.

[3] Sauro Succi. *The Lattice Boltzmann Equation for Fluid Dynamics and Beyond.* Oxford Science Publications, 2001. (And references therein).

[4] P. Lallemand Y.H.Qian, D. d'Humireres. Lattice bgk models for navier-stokes equation. *Europhys. Lett*, 17(6):479, 1992.

[5] L.D. Landau and E.M. Lifshitz. *Fluid Mechanics, Course of Theoretical Physics*, volume 6. Pergamon Press, 1966.

[6] M.Latva-Kokko and D.H. Rothman. Diffusion properties of gradient-based lattice boltzmann models of immiscible fluids. *Phys. Rev. E*, 71(056702), 2005.

[7] Guangdong Zhu O.Berk Usta, Alexander Alexeev and Anna C. Balazs. Modeling of microcapsules that communicate through nanoparticles to undergo self-propelled motion. *ACS NANO*, 2(3):471–476, 2008.

[8] Rolf Verberg Alexander Alexeev and Anna C. Balazs. Modeling the motion of microcapsules on compliant polymeric surfaces. *Macromolecules*, 38:10244–10260, 2005.

[9] J.H. Kinney A.J.C. Ladd. Elastic constants of cellular structures. *Physica A*, 240(1):349–360, 1997.

[10] G.K. Batchelor. *Introduction to Fluid Dynamics.* Cambridge Mathematical Library Series. Cambridge University Press, 2002.

[11] D.J. Tritton. *Physical Fluid Dynamics.* Oxford University Press, 1988.

[12] J. Happel and H. Brenner. *Low Reynolds Number Hydrodynamics: with special applications to particulate media (Mechanics of Fluids and Transport Processes).* Springer, 1983.

[13] G. Arfken. *Mathematical Methods for Physics*, volume 3. FL:Academic Press, 1985.

[14] Lars Onsager. Crystal statistics. i. a two-dimensional model with an order-disorder transition. *Phys. Rev*, 65:117–149, 1944.

[15] J.Hardy and O. de Pazzis. Molecular dynamics of a classical lattice gas: Transport properties and time correlation functions. *Phys. Rev. A*, 13:1949–1961, 1976.

[16] Uriel Frisch *et al.* Lattice gass hydrodynamics in two and three dimensions. *Complex Systems*, 1(4), 1987.

[17] P.J. Hoogerbrugge and J.M.V.A. Koelman. Simulation microscopic hydrodynamic phenomena with dissipative particle dynamics. *Europhysics. Lett*, 19(3):155–160, 1992.

[18] J.M.V.A. Koelman and P.J. Hoogerbrugge. Dynamic simulations of hard-sphere suspensions under steady shear. *Europhysics. Lett*, 21:363–368, 1993.

[19] Burkhard Dunweg Thomas Soddemann and Kurt Kremer. Dissipative particle dynamics: A useful thermostat for equilibrium and nonequilibrium molecular dynamics simulations. *Phys. Rev. E*, 68:2332–2335, 2003.

[20] R.D. Groot and P.B. Warren. Dissipative particle dynamics: Bridging the gap between atomistic and messcopic simulation. *J. Chem. Phys*, 107, 1997.

[21] W. R. Osborn M. R. Swift and J. M. Yeomans. Lattice boltzmann simulation of nonideal fluids. *Phys. Rev. Lett.*, 75:830 – 833, 1995.

[22] W. R. Osborn M. R. Swift, E. Orlandini and J. M.Yeomans. Lattice boltzmann simulations of liquid-gas and binary fluid systems. *Phys. Rev. E*, 54:5041–5052, 1996.

[23] I. Halliday M. M. Dupin and C.M. Care. Simulation of a microfluidic flow-focusing device. *Phys. Rev. E. Rappid Comm.*, 2006.

[24] S. Zaleski A. K. Gunstensen, D. H. Rothman and G. Zanetti. Lattice boltzmann model of immiscible fluids. *Phys. Rev. A*, 43:4320–4327, 1991.

[25] C. M. Care S. V. Lishchuk and I. Halliday. Lattice boltzmann algorithm for surface tension with greatly reduced microcurrents. *Phys. Rev. E*, 67(036701), 2003.

[26] E. P. Gross P. L. Bhatnagar and M. Krook. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. *Phys. Rev.*, 94:511–525, 1954.

[27] S. Chen G.D. Doolen S. Hou, Q. Zou and A.C. Cogley. A improved incompressible lattice boltzmann model for time-independent flows. *J. Stat. Phys.*, 81(1-2):35–48, 1995.

[28] C.M.Care K.Good I.Halliday, L.A.Hammond and A.Stevens. Lattice boltzmann equation hydrodynamics. *Phys. Rev. E*, 64, 2001.

[29] R. L. Liboff. *Kintetic Theory, Classical, Quantum, and relitavistic descriptions.* Springer–Verlag, 2003.

[30] S. Chen G. Doolen S. Hou, Q. Zou and A. C. Cogley. Simulation of cavity flow by the lattice boltzmann method. *J. Comp. Phys.*, 118(2):329–347, 1995.

[31] Zou Q and He X. On pressure and velocity boundary conditions for the lattice boltzmann bgk model. *Phys. Fluids*, 9(6):1591–1598, 1997.

[32] Rutherford Aris. *Vectors, Tensors and the Basic Equations of Fluid Mechanics.* Dover Publications, 1990.

[33] X. W. Shan and H. D. Chen. Simulation of nonideal gases and liquid-gas phase transitions by the lattice boltzmann equation. *Phys Rev. E*, 49:2941–2948, 1994.

[34] E. Aurell M. Do-Quang and M. Vergassola. An inventory of lattice boltzmann models for multiphase flows. Technical Report 00:03, Parallel and Scientific Computing Institute, Royal Institute of Technology and Uppsala University, September 2000.

[35] D. B. Kothe J. U. Brackbill and C. Zemach. A continuum method for modeling surface tension. *J. Comput. Phys.*, 100(2):335–354, 1992.

[36] A. W. Lees and S.F. Edwards. The computer study of transport processes under extreme conditions. *J. Phys C*, 5:1921–1928, 1972.

[37] A. J. Wagner and J. M. Yeomans. Phase separation under shear in two-dimensional binary fluids. *Phys. Rev. E*, 59:4366 – 4373, 1999.

[38] J. G. Georgiadis M. A. Gallivan, D. R. Noble and R. O. Buckius. An evaluation of the bounceback boundary condition for lattice boltzmann simulations. *Int. J. Num. Meth. Fluids*, pages 249–263, 1997.

[39] J. G. Georgiadis D. R. Noble, S. Chen and R. O. Buckius. A consistent hydrodynamic boundary condition for the lattice boltzmann method. *Phys. Fluids*, 7(1):203–209, 1995.

[40] L. S. Luo X. Y. He, Q. Zou and M. Dembo. Analytic solutions of simple flows and analysis of nonslip boundary conditions for the lattice boltzmann bgk model. *J. Stat. Phys.*, 87(1-2):115–136, 1997.

[41] R. Mei S. Chen, D. Martinez. On boundary conditions in lattice boltzmann methods. *Phys. Fluids*, 8(9):2527–2536, 1996.

[42] F. Ogino T. Inamuro, M. Yoshino. A non-slip boundary condition for lattice boltzmann simulations. *Phys. Fluids*, 7(12):2928–2930, 1995.

[43] In-Chan I. C. Kim. Second order bounce back boundary condition for the lattice boltzmann fluid simulation. *KSME Int. J.*, 14(1):84–92, 2000.

[44] P. A. Skordos. Initial and boundary conditions for the lattice boltzmann method. *Phys. Rev. E*, 48(6):4823 – 4842, 1993.

[45] I. Ginzbourg and D. d'Humières. Local second-order boundary methods for lattice boltzmann models. *J. Stat. Phys.*, 84(5-6):927–971, 1996.

[46] C. M. Care I. Halliday, L. A. Hammond and A. Stevens. Enhanced closure scheme for lattice boltzmann equation hydrodynamics. *J Phys A: Math. Gen*, 35(12), 2002.

[47] A.J.C Ladd and R.Verberg. Lattice-boltzmann simulations of particle-fluid suspensions. *J. Stat. Phys*, 104(5-6):1191–1251, 2001.

[48] Chuguang Zheng Zhaoli Guo and Baochang Shi. Discrete lattice effects on the forcing term in the lattice boltzmann method. *Phys. Rev. E*, 65, 2002.

[49] I. Halliday M.M. Dupin and C.M. Care. Multi–component lattice boltzmann equation for mesoscale blood flow. *J. Phys A: Math Gen*, 36, 2003.

[50] I. Halliday M.M. Dupin, T.J. Spencer and C.M. Care. A many-component lattice boltzmann equation simulation for transport of deformable particles. *Philos. Trans. R. Soc. London, Ser A*, 362(1822), 2004.

[51] Charles D. Eggleton Sameer Jadhav and Konstantinos Konstantopoulos. A 3-d computational model predicts that cell deformation affects selectin-mediated leukocyte rolling. *Biophys. J.*, 88(96), 2005.

[52] Asghar esmaeeli and Gretar tryggvason. Direct numerical simulations of bubbly flows. part 1. low reynolds number arrays. *J. Fluid. Mech*, 377:313–345, 1998.

[53] S. Sundaresan J. Lu K. Sankaranarayanan, I.G. Kevrekidis and G. Tryggvason. A comparative study of lattice boltzmann and front-tracking finite-difference methods for bubble simulations. *Int. J. Multiphase Flow*, 29(1):109–116, 2003.

[54] A. Wagner. The origin of spurious velocities in lattice boltzmann. *int. J. Mod. Phys. B*, 17:193, 2003.

[55] S. P. Thompson I. Halliday and C. M. Care. Macroscopic surface tension in a lattice bhatnagar-gross-krook model of two immiscible fluids. *Phys. Rev. E*, 57:514–523, 1998.

[56] A. J. C. Ladd. Numerical simulations of particulate suspensions via a discretized boltzmann equation. part 1. theoretical foundation. *J. Fluid. Mech*, 271:285–309, 1994.

[57] M. Cieplak R. Rybka U D'Ortona, D. Salin and J. Banavar. Two-color nonlinear boltzmann cellular automata: Surface tension and wetting. *Phys. Rev. E*, 51(4):3718 – 3728, 1995.

[58] M. Latva-Kokko and D.H. Rothman. Diffusion properties of gradient-based lattice boltzmann models of immiscible fluids. *Phys. Rev. E*, 71(056702), 2005.

[59] M. Henon. Viscosity of a lattice gas. *Complex Syst*, 1, 1987.

[60] P.B. Warren. Electroviscous transport problems via lattice-boltzmann. *Int. J. Mod. Phys. C*, 8(4):889 – 898, 1997.

[61] T. Reis and T. N. Phillips. Modified lattice boltzmann model for axisymmetric flows. *Phys. Rev. E*, 75(056703), 2007.

[62] R.G. Cox. The dynamics of the spreading of liquids on a solid surface. part 1. viscous flow. *Journal of Fluid Mechanics*, 168:169–194, 1986.